Assignment 2
A mathematical essay on Logistic regression.

Anik Bhowmick
Inter Disciplinary Dual Degree Data-Science
Indian Institute of Technology Madras
ae20b102@smail.iitm.ac.in

***Abstract*—This assignment discusses the application of a classification algorithm named Logistic Regression on the famous Titanic dataset. On April 15, 1912, the Titanic sank after colliding with an iceberg on its first voyage. More than 50% people died in this unfortunate incident. However, some groups of people survived. By studying the data and fitting the model, our task is to unfold those patterns that link the survival rate of the passenger to their various features such as age, gender, financial background, solo or accompanied traveller, etc.**

***Index Terms*—Feature selection, visualization, Logistic Regression, Sigmoid Function, Confusion Matrix, Accuracy, Precision, Recall, F_1 score, ROC.**

I. Introduction

- This assignment is a simple binary classifier about the survival of the onboard passengers of the wrecked Titanic ship using logistic regression. Although most onboard people died, certain patterns tell passengers of some particular class, women, children, etc., survived more than any other passengers. So, in this assignment, we will not only reveal that pattern but also demonstrate the working of logistic regression, which is capable of predicting, given some attributes of a passenger, whether or not that passenger survives. For this, we will also highlight some visualization techniques to assess the data and validate the model.
- Logistic regression is a supervised machine learning algorithm mainly used for classification tasks where the goal is to predict the probability that an instance belongs to a given class or not. It is a statistical algorithm that analyzes the relationship between a set of independent variables and the dependent binary variables. It is a powerful tool for decision-making. It's referred to as regression because it takes the output of the linear regression function as input and uses a sigmoid function to estimate the probability for the given class. Just like linear regression here, some cost function is used. The model parameters are trained by minimizing this cost function. Logistic regression can be easily distinguished from linear regression by looking at the target variable of the data. In the case of linear regression, the output is continuous. Whereas for logistic regression, it is discrete/ categorical (0-1 or yes-no type). But logistic regression can also be extended to multiclass classifiers using an algorithm called one vs. all. Here, the model predicts the probability for each class. The given data point is thought to have belonged to a class in which the model outputs the highest probability. However, in this assignment, we will restrict our discussion to binary classification problems only.
- The main task in this assignment is to build a classifier model that tries to find the relation between the survival of a passenger and their various features, such as which passenger class he is from. Male or female, travelling single or with family? Which age group does he belong to? And so on. To do so, we will demonstrate various data visualization and preprocessing techniques.
- This paper will demonstrate the data analysis technique relevant to logistic regression with the help of visual plots and mathematical equations. We will discuss various data handling processes such as non-numeric quantities, missing values, feature imputations, etc.

II. Logistic Regression

Logistic Regression is a supervised machine learning algorithm that first computes the continuous value output like linear regression and then converts it to a discrete variable target value using a logistic function or sigmoid function. So, it is used for predicting the categorical dependent variable using a given set of independent variables. All its properties are listed briefly below.

- Logistic regression predicts the output of a categorical dependent variable. It can be either Yes or No, 0 or 1, true or False, etc. Actually, it gives the probabilistic values between 0 and 1. We use some threshold of our own (0.5 by default), and if the probability is above 0.5, then it belongs to a particular class. If it is less, then the data belongs to another class.
- In logistic regression, instead of fitting a regression line, we fit a sigmoid function. This function is bounded above by 1 and bounded below by 0.

A. Assumptions

- Each observation is linearly independent of the other. So, there is no collinearity between any input variables.

- The target variable should be binary for the sigmoid function to work. For multiclass classification, a softmax function or one-vs-all algorithm can be used.
- There should be no outliers in the dataset.

B. Sigmoid function

Below is the sigmoid function.

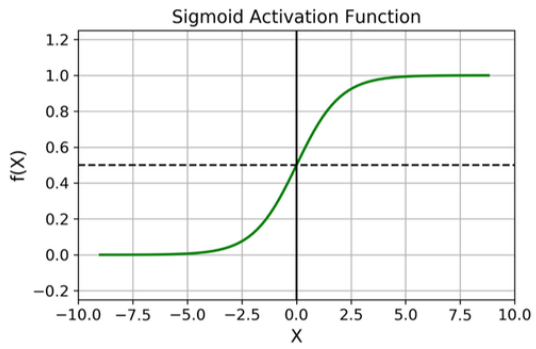

Fig. 1. Sigmoid function

The equation is :

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Some important features of this function are:

- $\sigma(z) \to 1$ as $z \to \infty$.
- $\sigma(z) \to 0$ as $z \to -\infty$.
- $\sigma(z)$ is bounded between 0 and 1.
- Probabilities are related to sigmoid function as $P(y = 1) = \sigma(z)$ and $P(y = 0) = 1 - \sigma(z)$.

C. Hypothesis function

Here, the hypothesis function is a sigmoid function applied to the linear function as given below.

$$log\left[\frac{h(x^{(i)};\boldsymbol{\theta})}{1 - h(x^{(i)};\boldsymbol{\theta})}\right] = z = \theta_0 + \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)} + ... + \theta_n x_n^{(i)}$$

$$log\left[\frac{h(x^{(i)};\boldsymbol{\theta})}{1 - h(x^{(i)};\boldsymbol{\theta})}\right] = \theta_0 + \theta^T X^{(i)} = z$$

$$h(x^{(i)};\boldsymbol{\theta}) = \frac{1}{1 + e^{-(\theta_0 + \theta^T X^{(i)})}}$$

Where θs are the parameters that the model determines, X is a column vector containing features of ith example. θ_0 is the bias term. θ is a column vector of $\theta_1, \theta_2, \theta_2, ...\theta_n$.

D. Cost Function

The cost function given below is for logistic regression, also known as binary cross-entropy loss function.

$$\mathcal{L} = \frac{1}{m}\sum_{i=1}^{i=m}\left[y^{(i)} log\left(h(x^{(i)})\right) + (1 - y^{(i)}) log\left(1 - h(x^{(i)})\right)\right]$$

E. Gradient Descent

Just like linear regression, we also use a gradient descent algorithm to minimize the cost function here. The parameter updation follows the rule as given below.

$$\frac{\partial \mathcal{L}}{\partial \theta_0} = \frac{1}{m}\sum_{i=1}^{i=m}\left(y^{(i)} - h(x^{(i)})\right)$$

$$\frac{\partial \mathcal{L}}{\partial \theta_k} = \frac{1}{m}\sum_{i=1}^{i=m}\left(y^{(i)} - h(x^{(i)})\right) x_k \quad k \neq 0$$

There is a non-trainable parameter called learning rate α is used to update the parameters as follows

$$\theta_0 := \theta_0 - \alpha \frac{\partial \mathcal{L}}{\partial \theta_0}$$

$$\theta_k := \theta_k - \alpha \frac{\partial \mathcal{L}}{\partial \theta_k}$$

The choice of learning rate decides the convergence of the cost function. A larger learning rate helps faster convergence, and a smaller one slows the convergence rate. But keeping a very high learning rate causes parameters to blow out sometimes. So, a typical value of this hyperparameter is around 0.01-0.0001. Again, this depends on problems to problems and the choice of cost function as well. In other words, we have to experiment with several learning rate values to get the most optimal one.

F. Evaluation Metrics

There are several evaluation metrics available to evaluate a logistic regression model.

- **Confusion Matrix** A confusion matrix is the table often used to describe the performance of a classification model on a set of test data for which the true values are known. A confusion matrix looks exactly as given below.

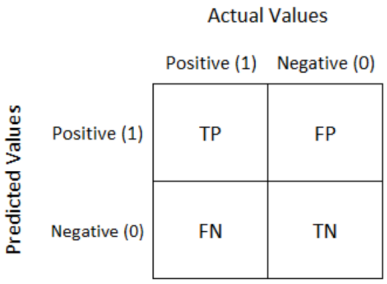

Fig. 2. Confusion matrix

TP: True positive is how many positives are true as predicted by the model.
FP: False positive is how many are predicted to be positive but not positive.
FN: False negative is how many are falsely predicted to be negative.
TN: True negative is how many are predicted to be negative, which are actually negative.

It is clear that only TP and TN are the correct predictions made by the model. The rest are wrong predictions. A good model should have FP and FN as small as possible.

- **Accuracy** We define accuracy as the fraction of correct prediction out of the total prediction given by the model.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

A good model should have high accuracy. But that is not a necessary condition. Even a highly accurate model can give more wrong predictions if it is trained on a highly imbalanced dataset(i.e., the number of one class instances is much larger than the other class). So, accuracy alone can not help us decide whether a model is good or not.

- **Precision**: This is defined as how many are actually positive out of total positive prediction.

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall**: or true positive rate is defined as the number of predicted positive out of actual positive. It is also known as Sensitivity.

$$\text{Recall (TPR)} = \frac{TP}{TP + FN}$$

Increasing recall decreases precision, often known as "precision-recall trade-off". It depends from problem to problem what we want: more precision or more recall. Like in the medical field of cancer disease detection models, we want the model to become robust to detect cancer. In this case, predicting a negative, even if a patient is positive, is highly undesirable. So, we want a more positive rate, meaning high recall. But in a problem like the Titanic dataset, we don't need our model to be that stringent in making decisions.

- $\boldsymbol{F_1}$ **score** : Is the harmonic mean of precision and recall.

$$\boldsymbol{F_1}\ \textbf{score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

If we don't want to emphasise the precision and recall individually, we check for a high F_1 score for a good model.

III. The Data

Brief information about the data:

- Total data points is 891. Columns are PassengerId, Survived, Pclass, Name, Sex, Age, SibSp, Parch, Ticket, Fare, Cabin, Embarked.
- Categorical columns are Name, Sex, Ticket, Cabin, and Embarked.
- Some columns have missing entries, such as Age and Cabin.

Variable	Definition	Key
survival	Survival	0=No, 1=Yes
pclass	Ticket class	1 = 1st, 2 = 2nd, 3 = 3rd
sex	Sex	M / F
Age	Age 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 of Embarkation	C = Cherbourg, Q = Queenstown, S = Southampton

TABLE I
THE DESCRIPTION OF THE COLUMNS OF THE DATA

IV. The Problem

As mentioned, our goal in this assignment is to predict whether a passenger survives or not in the Titanic accident by logistic regression. We followed the following steps to prepare the data before feeding it into the model. After training, we evaluated the model on several classification metrics. Then, we used the model to predict the test data set.

A. Cleaning and preparing the data

The dataset had some missing and categorical entries. In either case, the logistic regression model can not work. Access the dataset here

1) Handling the categorical columns: Some of the columns of the given dataset had categorical features such as "Sex", "Embarked", and "PClass". In Pclass, it is actually ordinal, so we have to consider it as categorical data. Because the model does not know it is ordinal data. It is just some number to it. But it can give more importance to larger numbers than smaller numbers or vice versa. To avoid this ambiguity, we will convert this also to one hot encoded vector. The table below explains how one hot encoding really works.

Emabarked	**Embarked_C**	**Embarked_Q**	**Embarked_S**
S	0	0	1
C	1	0	0
S	0	0	1
S	0	0	1
S	0	0	1

TABLE II
ONE HOT ENCODED FEATURE VECTOR FOR EMBARKED COLUMN FOR THREE PORTS

So, we see for the Emabarked S case, we get a vector representation (0,0,1). For C, it is (1,0,0), and for Q, it is (0,1,0). This kind of data representation is one hot encoded representation of the Emarked column. The same goes for Sex and PClass.

2) Merging some columns to single column: We see that survival is comparatively higher for people travelling with families having 1-2 family members. At the same time, passengers travelling alone died more in number.

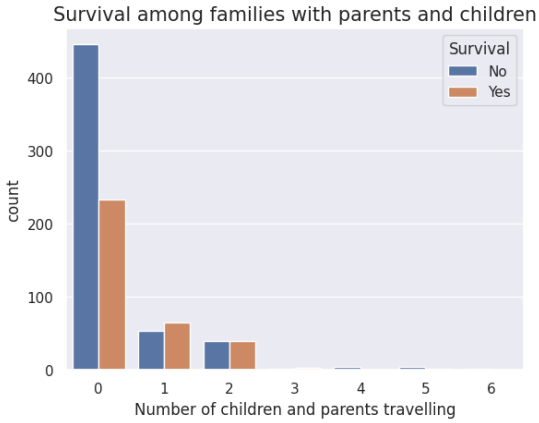

Fig. 3. Passengers travelling with parents and children

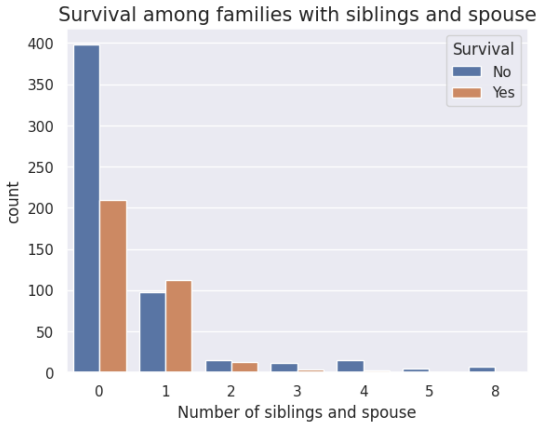

Fig. 4. Passengers travelling with siblings and spouse

So, instead of considering these two columns separately, we can merge them into a single column with the name 'Number of Family members'. The final bar plot looks as below.

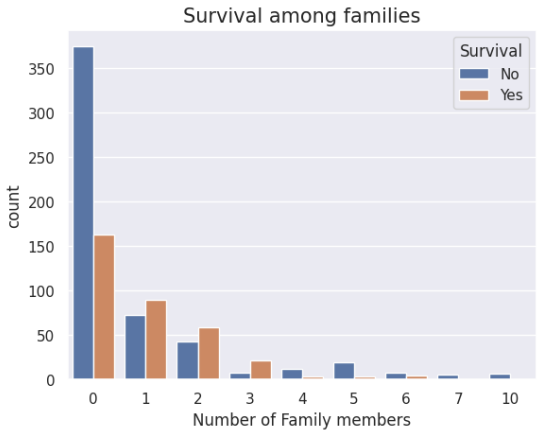

Fig. 5. Merged column

This plot shows that passengers with 1-3 family members survived more than others.

3) Handling the Missing values: As already mentioned, the "Age", "Cabin", and "Embarked column" has missing entries.

- In the **"Embarked"** column, we replaced the missing value with the port having the highest number of passengers boarded. In this case, it is Southampton.

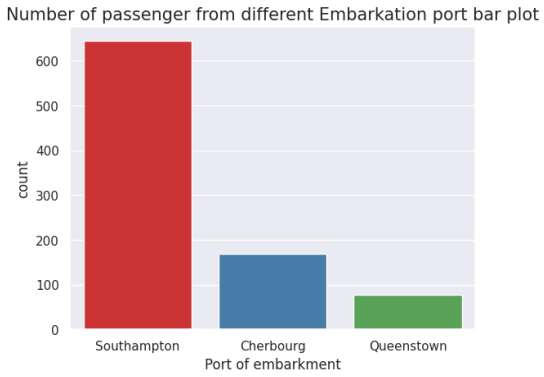

Fig. 6. Bar plot showing the number of passengers embarked from three ports

- For **"Age"** column, when we visualized the data distribution, it turned out to be right skewed. In the case of skewed data, we impute the missing values with the median.

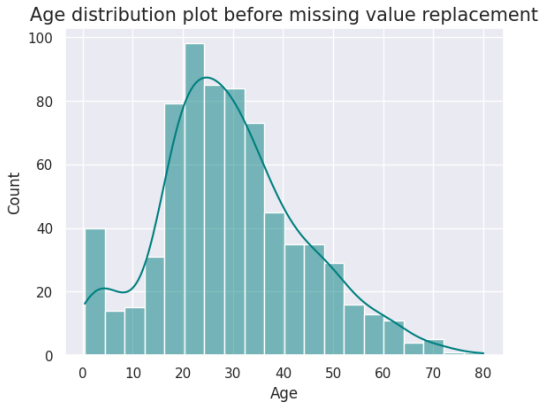

Fig. 7. Distribution of Age

But we didn't go by mere intuition and replace the missing entries with the median. Instead, we fitted the data, excluding the target, with a supervised algorithm called K Nearest Neighbors. We won't go into the detailed working of KNN in this assignment as this assignment is meant to focus on only logistic regression. We need to know that we can use the KNN imputation technique to replace missing values if the column is numerical. KNN uses neighbouring data points to decide what value to put at the missing place. After KNN imputation, the distribution looked as given below.

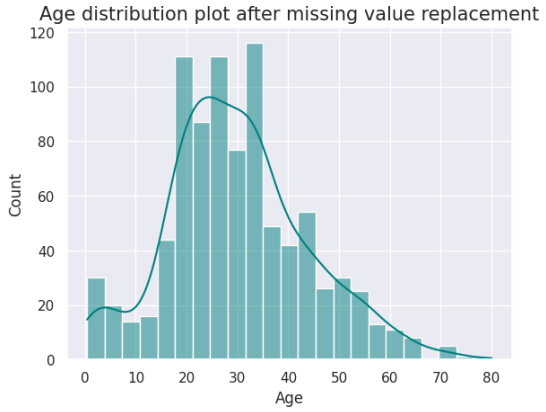

Fig. 8. Distribution of Age after missing value imputation

- In **"Cabin"** column, we saw 687 out of 891 data are missing. So, we decided to drop the entire column. Because with such a low number of data points, useful insight to replace missing values can not be drawn.

B. Exploratory analysis

1) Visualizing the correlation among features: We also checked the correlation plot beforehand. Because having near-linearly dependent data causes the model to make wrong decisions. However, it was less correlated even before missing data handling. Further, the correlation is removed because we already merged two columns, "Parch" and "SibSp" :

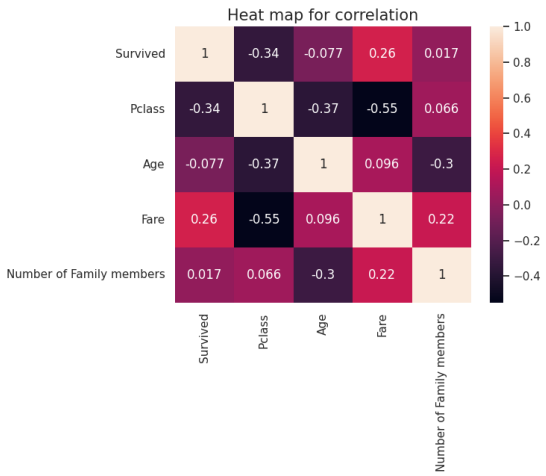

Fig. 9. Correlation plot

C. Splitting the Training data to Train and validation sets

To validate the model, we decided to split the model into training and validation sets. However, splitting is not a good idea with such low data samples. So, we decided to keep the validation set fraction as small as 0.15 of total data, i.e., 134 roughly.

D. Statistical model

Our model is a logistic regressor from sklearn. On the first model training, we achieved a fairly high accuracy on the validation data set 0.84. So we decided not to continue further with the training process. In fact, with this low number of data, we can't expect the model to perform better than this. From the output of the model, the statistical expression of the model is as follows. Here x_i is i the feature of a particular example.

$$z = -0.6112 - 0.5060x_1 + 0.0816x_2 - 0.3366x_3 + 0.5120x_4 + 0.1055x_5 - 0.5283x_6 + 0.0456x_7 + 0.0676x_8 - 0.0828x_9 - 1.2489x_{10}$$

$$h(x;\boldsymbol{\theta}) = \frac{1}{1+e^{-z}}$$

E. Visualization and validation

Below are the evaluation metrics for the validation set: We

Survival	Precision	Recall	F1-score
0	0.86	0.90	0.88
1	0.79	0.72	0.76

see our F_1 score for the survival (Yes) class is 0.76, which is less than the died class 0.88. This is because the data set is slightly imbalanced due to more dead people than survived. For an imbalanced data set, this number is fair enough.

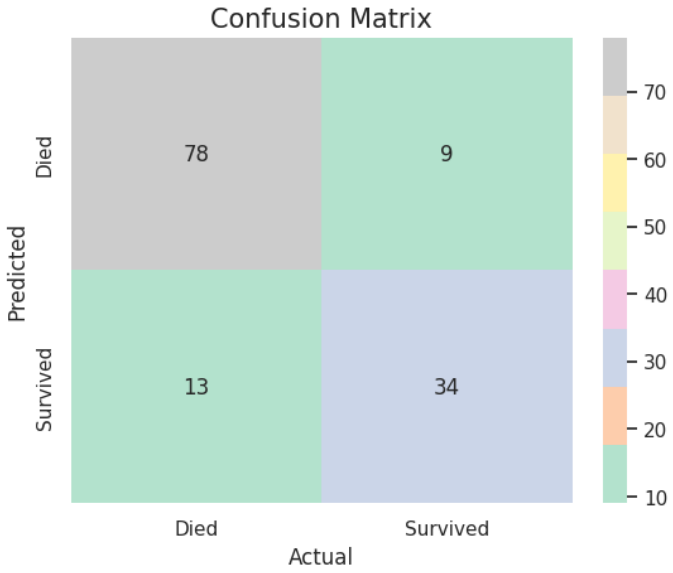

Fig. 10. Confusion Matrix

From this figure, we can infer the following

- **True Positive**=34 (Actual survived, predicted survived)
- **False Positive**=13 (Actual died, predicted survived)
- **False Negetive**=9 (Actual survived, predicted died)
- **True Negative**=78 (Actual died, predicted died)

From this, we get the accuracy as:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

$$\text{Accuracy} = \frac{34+78}{34+78+13+9} = 0.836$$

The goodness of a model can be tested by one more metric called Area under the ROC (Receiver operating characteristics) curve. The more the area, the better the model is. In our case, we achieved an AU-ROC of 0.87. Which indicates the model is performing well.

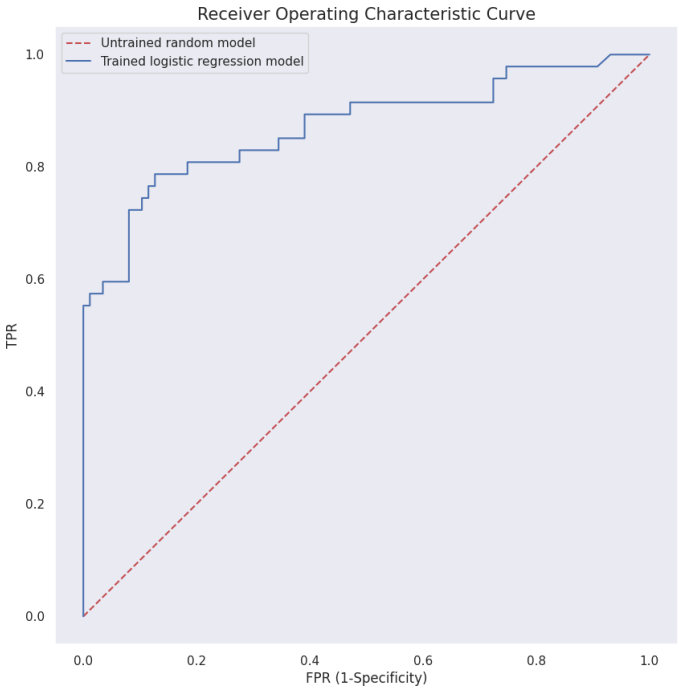

Fig. 11. Receiver operating characteristic curve

F. Insights captured by studying the data

After going through the training data, we captured some patterns. They are discussed below.

1) Insights in training data:

- The training data is a little imbalanced, with more deaths than survivals. The plot given below gives that comparison.

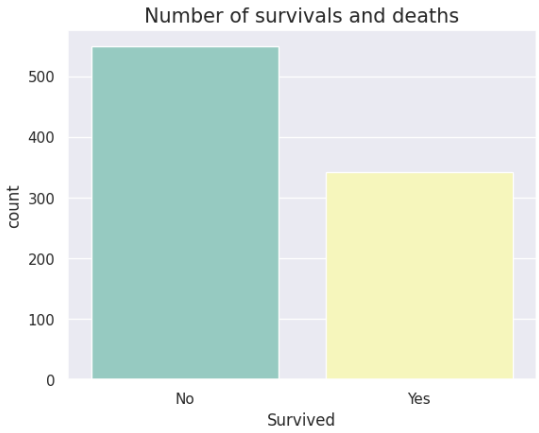

Fig. 12. Survival data

- We checked the survival rate among males and females. In this case, females are given more priority for their survival.

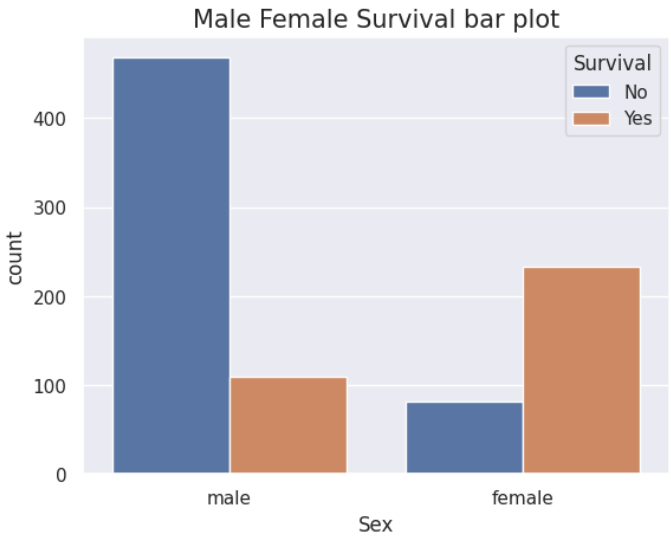

Fig. 13. Survival rate among genders

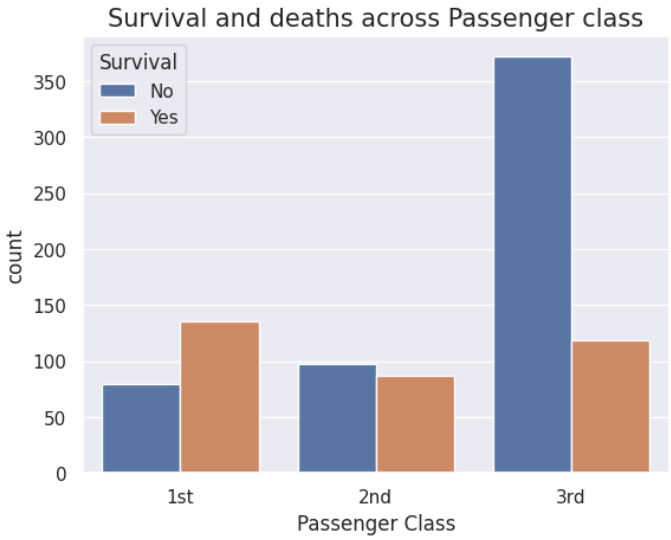

Fig. 14. Survival rate among passenger classes

- We further found that elite classes were prioritised for survival. So the survival rate among 1st class passengers is higher. So, this suggests money matters in terms of privileges.

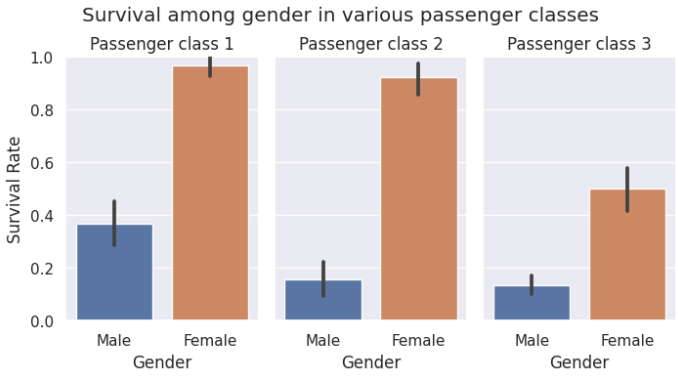

Fig. 15. Survival rate among genders in passenger classes

- From the above figure, it is clear females survived more than males across all passenger classes.
- We also checked the distribution of survival cases across various age groups. We found out that although death cases are almost equal for all age groups, younger people, especially those under 10, have a slightly higher survival rate.

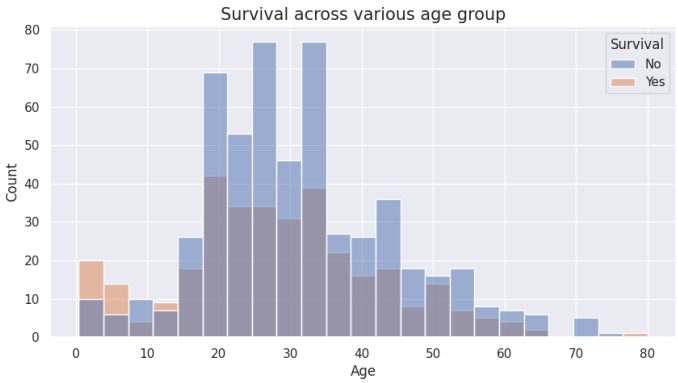

Fig. 16. Survival rate among various age groups

- Also, from Fig 5, it is clear that people with family survived more than single people.

2) Insights in testing data: We expect similar patterns in the predicted test data set. This will ensure the model is behaving consistently both on seen data as well as on unseen data.

- In the prediction on the test data, we obtained 265 people died, and 153 people survived. So, the survival-to-death ratio is approximately the same as in the train dataset.
- Here too people with family members survived more.

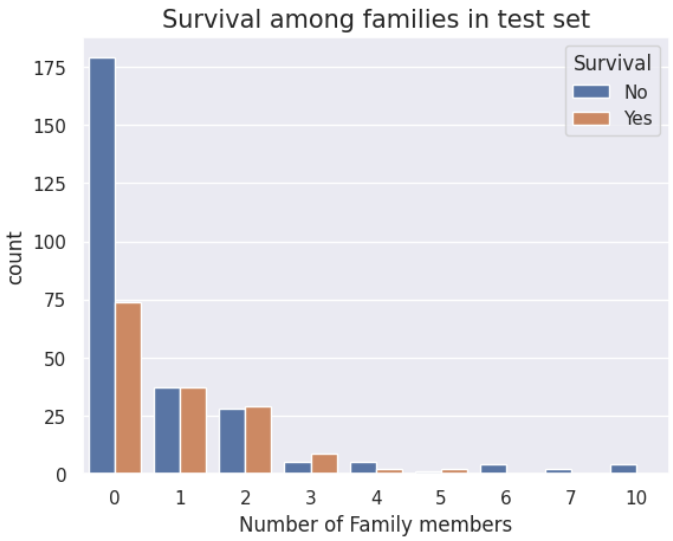

Fig. 17. Survival rate for people travelling with family in the test set.

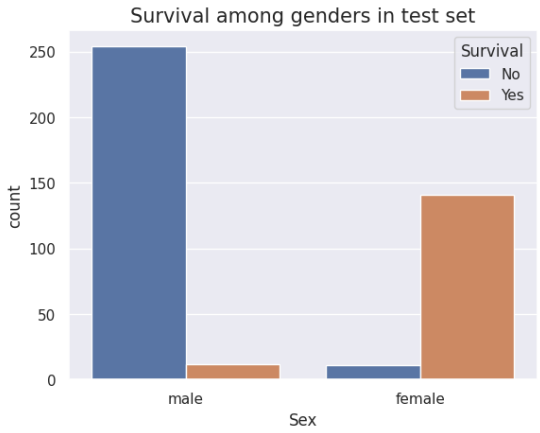

Fig. 18. Gender survival rate across passenger classes

- Again, here too, females survived more. In fact, no males from other than 1st class survived at all.

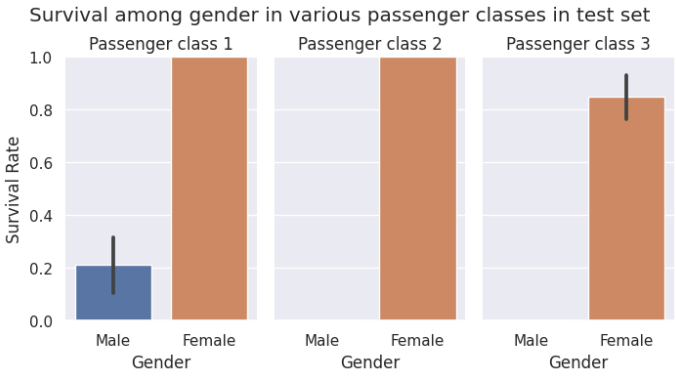

Fig. 19. Gender survival rate across passenger classes

- The Age group here under 5 has a higher survival rate.

It may be because the data set is fairly small, so it didn't properly keep information for all possible ages. Still, we can certainly state younger people have more survival than older ones in the test set too.

Fig. 20. Survival rate among age groups.

So, by observing the training data, we can infer the following in terms of survival:

- Females and children were given higher priority for survival.
- People with family members survived more.
- Passengers travelling in the first class survived more due to the fact that they were a highly privileged group. In fact, a large fraction of the male survival came from the 1st class only.

This inference we drew with respect to the ground truth labels. But our model also captured this trend, giving us strong confidence in our inference and intuition about the data. This is the only reason we found similar patterns in the test set. Examples include more female and younger aged people survivals, male survivals from 1st class in the majority, etc. So, undoubtedly, the model is performing really well.

V. Conclusions

Logistic regression is a powerful statistical method for classification-based modelling when the target variables are discrete. In this assignment, we demonstrated the techniques of handling tabular data and fitting a classification model using logistic regression. This assignment allowed us to explore the mathematical details of logistic regression and other statistical techniques. In fact, the fundamental mathematics of logistic regression is followed by neural networks. One big similarity is the activation function. In neural networks, too, we use the sigmoid function many times. There are certain limitations in the logistic regression, such as it assumes linear dependence between independent variables and the log odds of the dependent variable. But logistic regression won't perform well if this condition is not true in the data. Logistic regression is, in fact, a very simple model as compared to decision trees and neural networks. It often fails to capture complex patterns in the data. It is prone to overfitting when working with high-dimensional data and is highly sensitive to outliers. Perhaps using Gradient-boosting techniques such as XGBoost and AdaBoost would give much higher accuracy. Despite all these limitations, logistic regression is still a widely used powerful classifier model due to its simplicity and easy understandability.

References

[1] Logistic Regression: https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc

[2] Maths behind logistic regression: https://www.geeksforgeeks.org/understanding-logistic-regression/

[3] Evaluation Metrics: https://www.analyticsvidhya.com/blog/2021/07/metrics-to-evaluate-your-classification-model-to-take-the-right-decisions/

[4] Scikit learn KNN imputer: https://scikit-learn.org/stable/modules/generated/sklearn.impute.KNNImputer.html

[5] Scikit learn logistic regression: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

Access the original code here