An Investigation into the Interpretability of a Neural Network for Titanic Survival Prediction

Oscar Laurens

Introduction

It is well-known that linear models such as Linear and Logistic Regression (LR) are widely used for many different problems due to their simplicity and low cost, providing satisfactory results for relatively low effort. A core strength of these models are their global coefficients, which provide a straightforward, direct interpretation of how each feature in a dataset influences the model's prediction. The high level of explainability that linear models provide is a major reason why they are still so widely used despite the development of increasingly complex algorithms that not only perform better, but excel with learning non-linear relationships, as real-world data is rarely perfectly linear. These more sophisticated models, like neural networks, are often referred to as 'black box' models due to the large amount of calculations that take place to predict an output. While predictive performance is significantly better, we lose the ability to understand why they made the predicition they did. In industries such as medicine, knowing why a model made a certain diagnosis is equally as important, if not more, than predictive accuracy, due to the decision potentially having significant impact on a patient's health. A doctor justifying a diagnosis by saying "the model said so", is not ethically or even legally justifiable. Another reason is no model is correct 100% of the time, so when a model makes a mistake, it is not always understood why, which can make fixing these problems more difficult.

The field of ExplainableAI (XAI) aims to create solutions to this significant issue through providing methods or systems to explain model behaviour. This paper investigates a simple, local method that takes advantage of Rectified Linear Unit (ReLU) activations in neural networks, and compares the explanations extracted to the global explanations provided by the coefficients of a Logistic Regression model.

Data

Feature	Meaning
pclass	Passenger class (1 = 1st, 2 = 2nd, 3 = 3rd)
sex	Sex (male, female)
age	Age
sibsp	Number of siblings/spouses aboard
parch	Number of parents/children aboard
fare	Passenger fare price
embarked	Port of embarkation (C = Cherbourg, Q = Queenstown), S = Southampton

The Titanic Surivial dataset is a well-known dataset often used as a benchmark dataset for classification modelling. The dataset contains 1309 passengers, which, while small, was sufficient for the small network architecture used to achieve satisfactory predictive performance with an 80:20 split for training and testing. Models were trained to predict whether a passenger survived (1) or died (0).

The 'name', 'ticket', 'boat', 'body' features were removed due adding unnecessary complexity to the dataset; this also meant that all models should be able to learn relationships to a high enough standard for further analysis. Missing values in continuous feature columns were filled in using the median value of the entire feature column, while missing categorical values were replaced with the most frequent values in the column. Finally, continuous features were scaled to have a median of 0 and a standard deviation of 1, to prevent features with large scales from dominating during model training and categorical values ('sex' and 'embarked') were one-hot-encoded.

Methodology

The neural network is a feed-forward network consisting of two hidden layers (32 and 16 neurons), with ReLU activation functions after each layer. This model was then trained for 150 epochs, using the Adam optimiser and Binary Cross Entropy Loss. The Logistic Regression model was trained with a regularisation strength of 0.5. All models were trained using a set seed of 42.

To interpret the neural network, we look at its behaviour in the local region of a specific input. The network is composed entirely of linear layers and ReLU activation functions. The non-linear activation function allows the model to learn non-linear relationships as its being trained, but when inference is ran on a single test input, an active path of neurons is created upon prediction, where the output of ReLU for a layer L is precisely the pre-activation for layer L. Therefore the model essentiallty computes a composition of linear functions, leading to a single, equivalent linear function that is valid for the local region surrounding the prediction.

Network Definition

Let the trained neural network be a function $f : \mathbb{R}^n \to \mathbb{R}$ with L hidden layers. For an input $x \in \mathbb{R}^n$, the output of the first hidden layer, l_1 is given by:

$$h_1 = \sigma(W_1 x + b_1) \tag{1}$$

where W_1 is the weight matrix, b_1 the bias vector for the first layer, and σ is the ReLU activation function defined as $\sigma(z) = max(0, z)$.

Then the output for any subsequent layer l_k is given by:

$$h_k = \sigma(W_k h_{k-1} + b_k) \tag{2}$$

The final output of the network is a linear transformation of the last hidden layer's output:

$$f(x) = W_{out} h_L + b_{out} \tag{3}$$

Local Linear Approximation

For a given input sample, x_0, each ReLU unit is either "on" (its input is positive) or "off" (its input is zero or negative). Therefore we can define a diagonal matrix, D_k, for each layer l_k where the diagonal elements are:

$$(D_k)_{ii} = \begin{cases} 1 & \text{if } (W_k h_{k-1} + b_k)_i > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

The diagonal matrix acts as a switch, removing the non-linear part of the ReLU function for the given input. By grouping all the weight and diagonal matrices, the equivalent weight matrix W_{eq}, that represents the sensitivity of the model's output to each input feature, can be found:

$$W_{eq} = W_{out} D_L W_L D_{L-1} W_{L-1} \dots D_1 W_1 \tag{5}$$

Similarly, all the bias terms from the "active" neurons are accumulated into a single equivalent bias vector, b_{eq}. This term is an aggregation of the original biases, each transformed by the weights of the subsequent layers. The non-linear function $f(x)$ can therefore be perfectly approximated in the local region of x_0 by the linear function

$$f(x_0) \approx W_{eq} x_0 + b_{eq} \tag{6}$$

for which W_{eq} provides the local feature contributions for the prediciton on a given input x_0, which we visualise in subsequent waterfall plots.

Results

This section presents the overall performance of the trained models as well as a detailed analysis of three representative case studies to compare their explanatory outputs.

Model Performance

Model	Accuracy	Precision (Survived)	Recall (Survived)	F1 Score (Survived)
Neural Network	**85.0%**	**0.83**	**0.75**	**0.79**
Decision Tree	83.0%	0.81	0.74	0.77
Logistic Regression	80.0%	0.76	0.71	0.74

All models show similar behaviours in their predictions on the Titanic Survival dataset - they were better in their confidence to predict survivors (Precision) than ensuring they identified all true survivors (Recall). A significant reason for this is because of the class imbalance in the dataset, where there were more deaths than survivals, which leads to all models being more cautious about predicting survival.

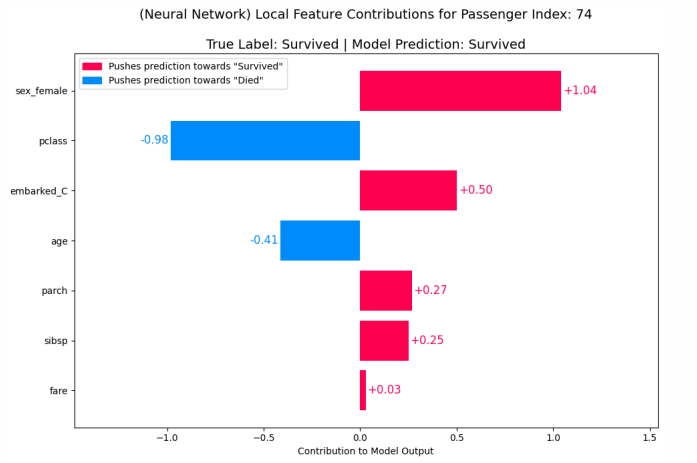
Case Study Analysis

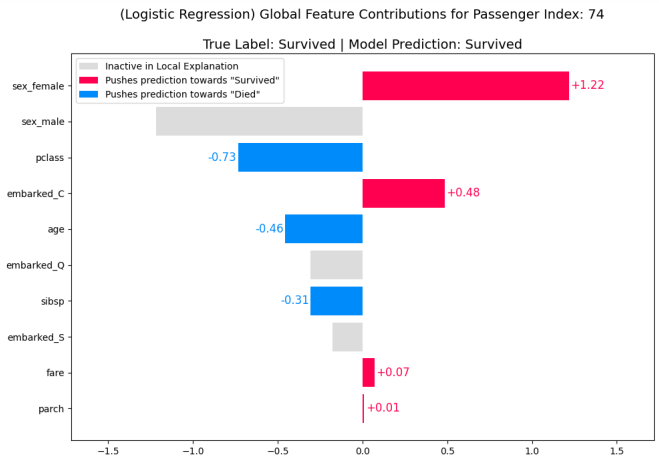
For each case study, a corresponding waterfall plot of the Logistic Regression's feature contributions is provided alongside the feature contributions of the neural network. It is worth pointing out that these two plots, while similar-looking, cannot be directly compared. Continous features in the Logistic Regression waterfall plot describe the effect of increasing the feature's value on its contribution to the final decision (e.g a feature contribution of -0.7 means increasing the value of the feature pulls the prediction towards predicting a death.). However, categorical features can be compared more directly, but it still stands that one coefficient is local while the other is global.

Case Study 1: Simple Agreement

This case study examines Passenger 74, an archetypal example where all models correctly predicted survival.

pclass	sex	age	sibsp	parch	fare	embarked
1	female	26.0	1	0	136.7792	C



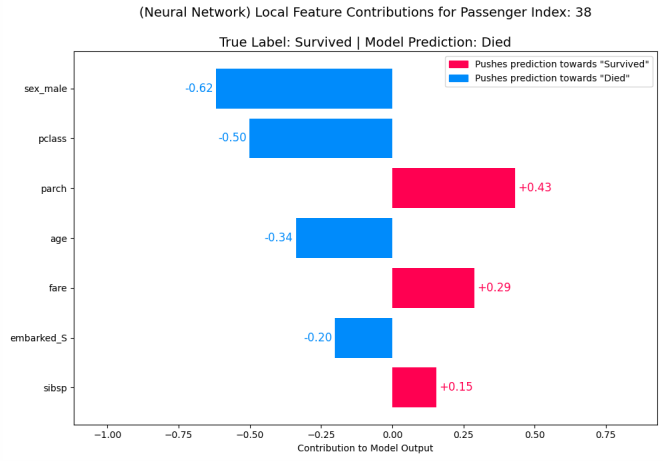


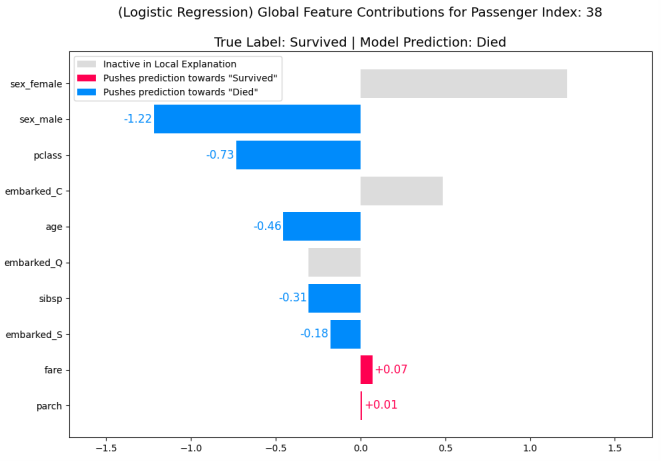
Both the Neural Network and Logistic Regression models showcase the same order of feature contributions (down to the final two features) towards predicting surivival, consequently outputting the same prediction. Both models agree on the effects of sex and port of embarkation on the probability of survival - female passengers generally have a high likelihood of survival and both models have demonstrated this. It is also worth pointing out that the LR model learns that passengers who embarked from Cherbourg have the highest surivial rates, while passengers who embarked from Southampton had the lowest. The neural network has potentially figured this out as well as an embarkation from Cherbourg has increased the probability of a prediction for survival. Both models also tie little significance to the fare price in relation to likelihood of survival.

Case Study 2: Incorrect Prediction

This case study examines Passenger 38, an example where both models fail to correctly predict the passenger's survival.

pclass	sex	age	sibsp	parch	fare	embarked
3	male	45.0	0	0	8.05	S

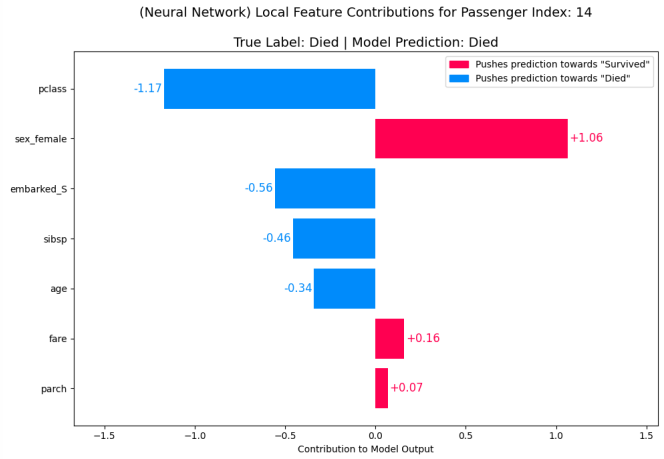


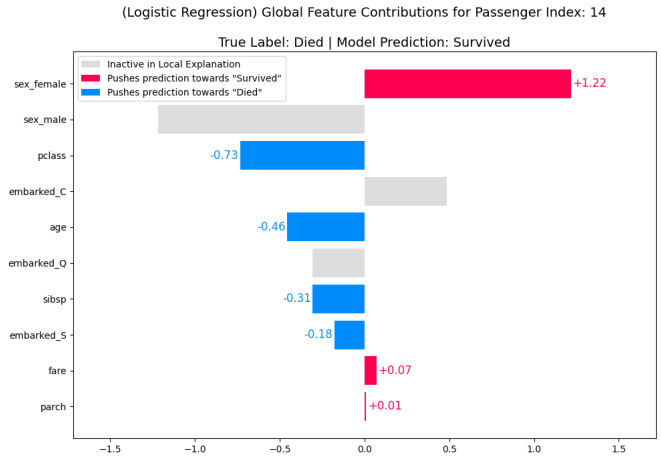


Both models incorrectly predict the passenger's death, but showcase different contributions for their incorrect decisions. The neural network in this case assigned a larger sigificance to the no. of parents/children abroad, along with the fare price, which pushed the prediction towards survival. For majority of the remaining features, the models seem to agree - the passenger was a 45 year old male in 3rd class, and each of these features is understood to decrease survival likehood by both models.

Case Study 3: Neural Nework outperforms LR

pclass	sex	age	sibsp	parch	fare	embarked
3	female	26.0	1	0	16.1	S





The neural network correctly predicts the passenger's death, while the LR model fails to do so. The fact that the passenger was female is a large factor when it comes to survival and both models portray this, being in the top 2 most influential feature contributions. The passenger embarked from Portsmouth, and had no parents or children on board, which are both considered significant negative contributions by the models. The difference in prediction then lies in the strength of the passenger's class on the final prediction, where the neural network has it as the strongest factor for the passenger having died, whereas the LR models sees it as a less significant than gender.

Discussion

All three case studies demonstrate the core strenght of a non-linear architecture like a neural network to that of the Logistic Regression model. When both models succesfully predicted survival, they had similar reasons for it. However for the other two cases, the difference in behaviours was larger. This difference comes down to the fact that the neural network is able to learn feature interactions, such that for every passenger it comes across, the combination of feature contributions can change depending on how different features relate to each other. A logistic regression model applies a single, global linear function to the whole dataset, and so is incredibly unlikely to correctly classify all datapoints in a non-linear dataset. This difference is showcased in Case Study 3 where despite the passenger being female, the neural network focuses more on the passenger's class which "edges out" the "blanket statement" of the Logistic Regression model, leading to only the neural network making a correct predicition.

Case Study 2 comes as a difficult example for both models, as the passenger was a 45-year old male in 3rd class. These 3 features, independently, were not common amongst passengers who survived, and so the fact that they were all present together for this passenger meant that, as expected, the models would most likely misclassify the passenger as a death. Cases such as these are ultimately difficult to classify by models and humans alike, and so more complex, non-linear architectures can also fail to classify this passenger correctly, emphasising the non-linear nature of real world data.

This analysis, while successful in demonstrating the method of Local Linear Approximation for comparison with a linear model, is incredibly limited to that of simple neural network architectures that consist of only ReLU activations. Furthermore, this method cannot be applied to architectures with additional mechanisms, such as Convolution Layers in Convolutional Neural Networks (CNNs) or the attention mechanisms in Transformers. In conclusion, this paper successfully demonstrated that a comparative analysis against a simpler, transparent model is an effective method for interpreting the local, non-linear behaviour of a neural network, but is limited to very specific conditions.