DESIGN A NEURAL NETWORK TO GENERATE A SURVIVAL CLASSIFIER FOR THE TITANIC DATASET

Nguyen H. Hoang, Pham X. Cuong

Le Quy Don Technical University, Vietnam

ABSTRACT

The sinking of the Titanic is one of the most infamous shipwrecks in history. On April 15, 1912, during her maiden voyage, the widely considered "unsinkable" RMS Titanic sank after colliding with an iceberg. Unfortunately, there weren't enough lifeboats for everyone onboard, resulting in the death of 1502 out of 2224 passengers and crew. In this report, we present a predictive model that answers the question: "what sorts of people were more likely to survive?" using passenger data (ie name, age, gender, socio-economic class, etc).

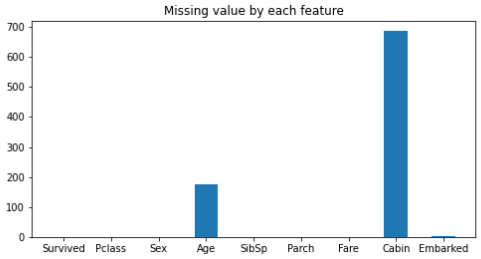
1. INTRODUCTION

In this report, we have gain access to two similar datasets that include passenger information like name, age, gender, socio-economic class, etc. One dataset is titled 'train.csv' and the other is titled 'test.csv'. Train.csv will contain the details of a subset of the passengers on board (891 to be exact) and importantly, will reveal whether they survived or not, also known as the "ground truth". The 'test.csv' dataset contains similar information but does not disclose the "ground truth" for each passenger. It's your job to predict these outcomes. Using a neural network for traning in the train.csv data, predict whether the other 418 passengers on board (found in test.csv) survived.

- The dataset have **11** features: *PassengerId*, *Pclass*, *Name*, *Sex*, *Age*, *SibSp*, *Parch*, *Ticket*, *Fare*, *Cabin*, *Embarked* and *Survived*. The *Survived* show that **2** labels Survived or not with **1** and **0**.
- While there was some element of luck involved in surviving, it seems some groups of people were more likely to survive than others. So we will figure out that with neural network.
- By using a Multilayer perceptron (MLP) neural network, we can predict the outcome of the survival of the passengers.
- In this report, we will show how to use the 'train.csv' dataset to train a neural network and then use the trained model to predict the outcome of the survival of the passengers in the 'test.csv' dataset.

2. METHOD

- Preprocessing the data:
  - In the dataset, we drop *PassengerId*, *Name*, and *Ticket*, Because they are unique by each passenger. We also drop *Cabin* because total of dataset have **891** samples and *Cabin* have **687** is **None**.



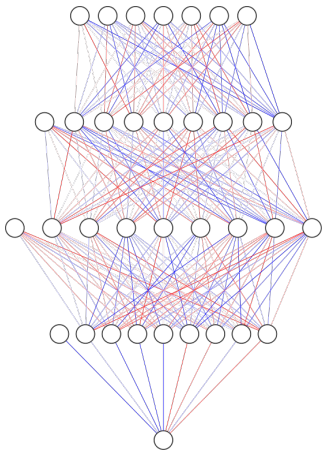
  - Normalize the data, we vectorize from discrete value to unique number value and use MinMaxScaler to normalize data.

$$X_{new} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

  - Impute the missing values in *Age* and *Embarked*, we use KNN algorithm to impute the missing values with **N neighbors** is 2 and formula caculate distance is **nan euclidean distances** .

$$NED(x, y) = \sqrt{\frac{S}{S_{nnan}} \sum_{i=0}^{n} d(x, y)^2}$$

  - Split the data into training, testing sets and validation sets with ratio is **7**:**2**:**1**.
- Build a Multilayer perceptron (MLP) neural network with the following hyperparameters:
  - **Number of input layers** is **7** from the input shape of the dataset after processed.
  - **Number of hidden layers** is **3** and **number of neurons** in each layer is **8** and activation function is **ReLU**.
  - **Number of output layers** is **1** for classification is survival or not with **Sigmoid** activation function.



– We use **Adam Optimization Algorithm** for update network weights iterative based on training data.

$$g_n \leftarrow \nabla f(\theta_{n-1}) \tag{1}$$

$$m_n \leftarrow \frac{\beta_1}{1-\beta_1^n} m_{n-1} + \frac{1-\beta_1}{1-\beta_1^n} g_n \tag{2}$$

$$v_n \leftarrow \frac{\beta_2}{1-\beta_2^n} v_{n-1} + \frac{1-\beta_2}{1-\beta_2^n} g_n \odot g_n \tag{3}$$

$$\theta_n \leftarrow \theta_{n-1} - \alpha \frac{m_n}{\sqrt{v_n} + \epsilon} \tag{4}$$

with α is **0.001**, ϵ is **1e-07**, β_1 and β_2 are **0.9** and **0.999** respectively.

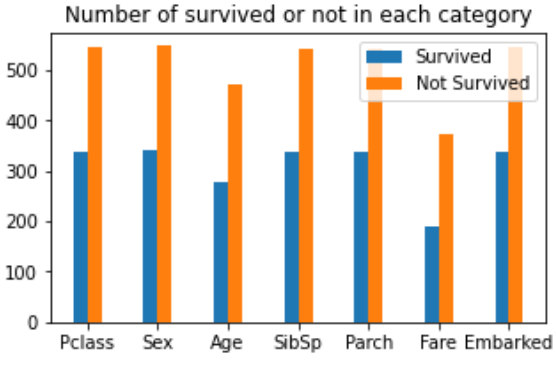
- This is a binary classification problem, so we use **binary cross-entropy** loss function.

$$-p(x).\log q(x) - (1-p(x)).log(1-q(x)) \tag{5}$$

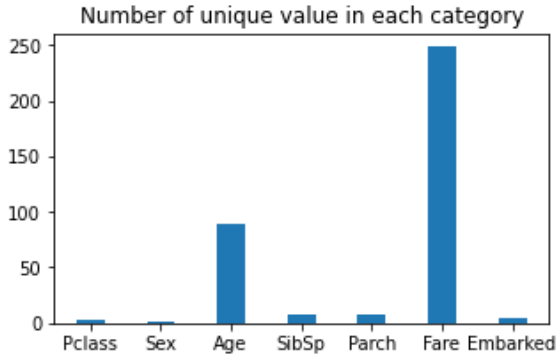
with $p(x)$ is target probability and $q(x)$ is predicted probability.

3. EXPERIMENTAL RESULTS AND ANALYSIS

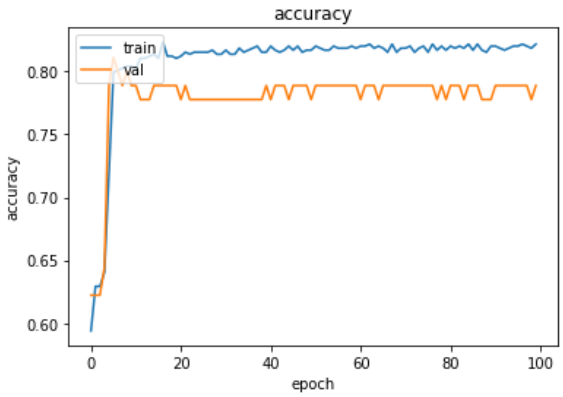
- The train dataset

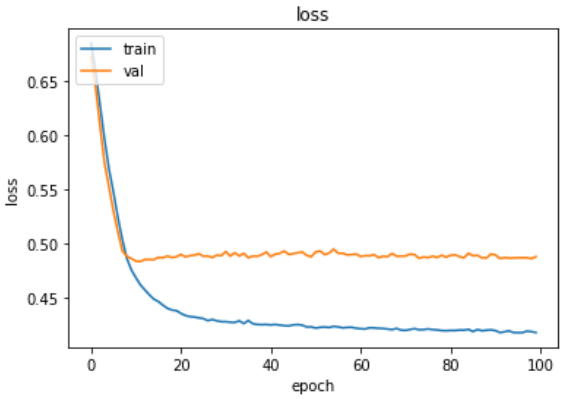


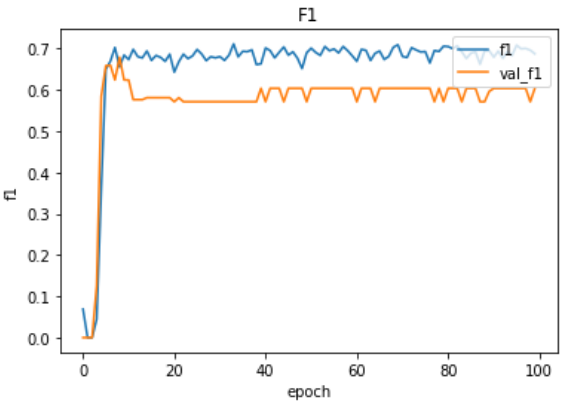
as we can see, there are 'Not survived' more than 'Survived' by each category.

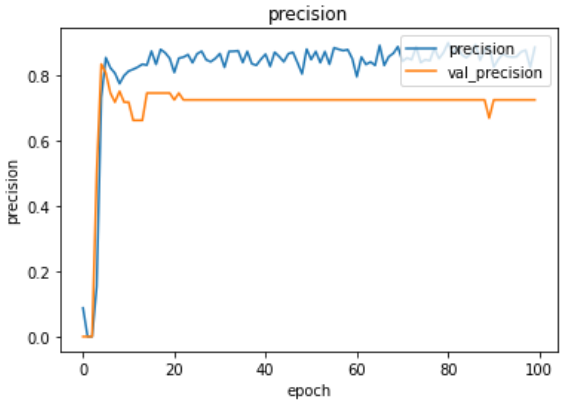


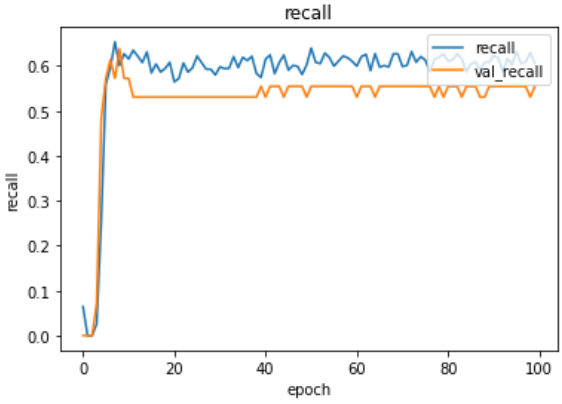
- We training with **100** epochs and **batch size** is **16**.
- There are the best accuracy, loss, F1 score, precision, and recall on train dataset and validation dataset after train 5 times.











- The results on test dataset.

Train times	1	**2**	3	4	5
Accuracy	0.803	**0.837**	0.808	0.808	0.820
Loss	0.448	**0.440**	0.445	0.442	0.454
F1-score	0.732	**0.778**	0.740	0.730	0.754
Precision	0.843	**0.899**	0.859	0.880	0.868
Recall	0.662	**0.694**	0.655	0.637	0.673

- Because Recall is low compared to Precision, it can be seen that the model is making the choice 'Not surviverd' over 'Surviverd'.
- Loss is quite high at 0.44, because the sigmoid activation function is curved, it is difficult to separate clearly between 0 and 1.

4. CONCLUSION

Using neural networks to solve the problem and achieve not bad results, but there will be machine learning algorithms that give higher accuracy. Therefore, it will be necessary to improve this neural network in different ways to give better results.