CS3120
MACHINE LEARNING AND NEURAL NETWORKS

Titanic: Machine Learning from Disaster
Assignment - Project Report

Faculty of Science, University of Colombo

K.K. de Silva - 2015s15400 - s12744
T.M. Hewavithana - 2015s15420 - s12764
W.G.A.S. Sandareka - 2015s15477 - s12819

CONTENT

INTRODUCTION

The RMS Titanic was a British passenger liner that sank in the North Atlantic Ocean in the early morning hours of 15th April 1912, after it collided with an iceberg during its maiden voyage from Southampton to New York City. There were an estimated 2,224 passengers and crew aboard the ship, and more than 1,502 died, making it one of the deadliest commercial peacetime maritime disasters in modern history. The Titanic was built by the Harland and Wolff shipyard in Belfast. Thomas Andrews, her architect, died in the disaster.

One of the reasons that the shipwreck led to such loss of life was that there were not enough lifeboats for the passengers and crew. Meanwhile, passengers and some crew members were evacuated in lifeboats, many of which were launched only partially loaded. A disproportionate number of men were left aboard because of a "women and children first" protocol for loading lifeboats. Although there was some element of luck involved in surviving the sinking, some groups of people were more likely to survive than others, such as women, children, and the upper-class.

In this project, the goal is to use the tools of machine learning to construct a model to predict which passengers survived the incident. Kaggle defines the problem to solve while providing the datasets for training the data science model and testing the model results against a test dataset for the Titanic Survival competition.

Using a training set of samples listing passengers who survived or did not survive the Titanic disaster, the neural network model determines if the passengers in the test dataset survived or not. Since input-target pairs of data are used in training, supervised machine learning approaches are employed. The data set contains information about the passengers such as age, sex and economic status which are important in building a correlational relationship between these variables and the probability of survival of the passengers.

Goal

To create a model that can predict the survivors of the Titanic shipwreck using machine learning tools.

Objectives

To learn about practical machine learning concepts.
To use tools of machine learning to construct models to predict the survival of passengers.

METHODOLOGY

The methodology is based on a Pattern Recognition and Classification Neural Network on the given dataset to predict survival rate of passengers of the Titanic shipwreck. The dataset contains 06 attributes which are assumed to be correlated with the survival rate. The data set was analyzed and processed to be compatible with the network model and to improve the model accuracy and the prediction model was built using the Neural Network Toolbox of Matlab. The predictions and the performance of the model were analyzed.

Data Set

The data has been split into two groups as training set (train.csv) and test set (test.csv). The training set is used to build your machine learning models, along with the target outcome. The test set is used to see how well the model performs on unseen data (without targets) and predict the survival of each passenger in the test set..

Variables;
Sex – Sex of the passenger

Age - Age of the passenger in years

pclass - Ticket class 1 = 1st, 2 = 2nd, 3 = 3rd

sibsp – Number of siblings / spouses aboard the Titanic

parch - Number 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]

Correlations:

- pclass can be considered as an indicator for socio-economic status [1st = Upper, 2nd = Middle, 3rd = Lower]. The higher classes were given priority in boarding the life boats.
- “women and children first" protocol for loading lifeboats gave higher survival rates for the females and ages below 18 years.
- sibsp and parch: The dataset defines family relations between siblings, spouses and parents and children. Passengers with family members tend to have a higher survival rate.

Data processing

- Original training data set

	A	B	C	D	E	F	G	H	I	J	K	L
1	Passenger	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
2	1	0	3	Braund, M	male	22	1	0	A/5 21171	7.25		S
3	2	1	1	Cumings, M	female	38	1	0	PC 17599	71.2833	C85	C
4	3	1	3	Heikkinen,	female	26	0	0	STON/O2.	7.925		S
5	4	1	1	Futrelle, M	female	35	1	0	113803	53.1	C123	S
6	5	0	3	Allen, Mr. '	male	35	0	0	373450	8.05		S
7	6	0	3	Moran, Mr	male		0	0	330877	8.4583		Q
8	7	0	1	McCarthy,	male	54	0	0	17463	51.8625	E46	S
9	8	0	3	Palsson, M	male	2	3	1	349909	21.075		S
10	9	1	3	Johnson, M	female	27	0	2	347742	11.1333		S
11	10	1	2	Nasser, Mr	female	14	1	0	237736	30.0708		C
12	11	1	3	Sandstrom	female	4	1	1	PP 9549	16.7	G6	S
13	12	1	1	Bonnell, M	female	58	0	0	113783	26.55	C103	S
14	13	0	3	Saunderco	male	20	0	0	A/5. 2151	8.05		S
15	14	0	3	Andersson	male	39	1	5	347082	31.275		S
16	15	0	3	Vestrom, M	female	14	0	0	350406	7.8542		S
17	16	1	2	Hewlett, M	female	55	0	0	248706	16		S
18	17	0	3	Rice, Mast	male	2	4	1	382652	29.125		Q

- Original Testing Data

	A	B	C	D	E	F	G	H	I	J	K	L
1	Passenger	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	
2	892	3	Kelly, Mr. J	male	34.5	0	0	330911	7.8292		Q	
3	893	3	Wilkes, Mr	female	47	1	0	363272	7		S	
4	894	2	Myles, Mr.	male	62	0	0	240276	9.6875		Q	
5	895	3	Wirz, Mr. A	male	27	0	0	315154	8.6625		S	
6	896	3	Hirvonen,	female	22	1	1	3101298	12.2875		S	
7	897	3	Svensson,	male	14	0	0	7538	9.225		S	
8	898	3	Connolly, M	female	30	0	0	330972	7.6292		Q	
9	899	2	Caldwell, M	male	26	1	1	248738	29		S	
10	900	3	Abrahim, M	female	18	0	0	2657	7.2292		C	
11	901	3	Davies, Mr	male	21	2	0	A/4 48871	24.15		S	
12	902	3	Ilieff, Mr. Y	male		0	0	349220	7.8958		S	
13	903	1	Jones, Mr.	male	46	0	0	694	26		S	
14	904	1	Snyder, Mr	female	23	1	0	21228	82.2667	B45	S	
15	905	2	Howard, M	male	63	1	0	24065	26		S	
16	906	1	Chaffee, M	female	47	1	0	W.E.P. 573	61.175	E31	S	

Training and Testing data were modified by;

- Name, Ticket number, fare and Port of embarkation were removed from the data set since they have no correlation with the probability of survival.
- Sex was converted to the binary format as 'Male' = 0 and 'Female' = 1
- Age was converted to binary format as if((Age>=18)?1:0) [Age not included in the data set was assumed to be an adult (i.e. 1)].
- Cabins of passengers were represented by numbers from 1 to 7 corresponding to A to G. [Cabin details not provided were given the data 0].

After modifications the target data set was separated from the training data set. The data was transposed to form a column matrix. The final data files were in the format of .csv (comma separated values) which is compatible with Matlab.

- Modified Training Data [6 x 891]

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	3	3	3	1	3	3	3	3	3	3	2	3	3	1
2	0	0	0	0	0	0	0	1	0	1	0	1	0	0
3	1	1	1	1	0	1	1	0	0	1	1	0	1	1
4	1	0	0	0	3	0	1	0	4	1	0	3	0	3
5	0	0	0	0	1	0	5	0	1	0	0	1	0	2
6	0	0	0	5	0	0	0	0	0	0	0	0	0	3
7														
8														

- Modified Targets of the Training Data set [1 x 891]

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2													
3													
4													
5													

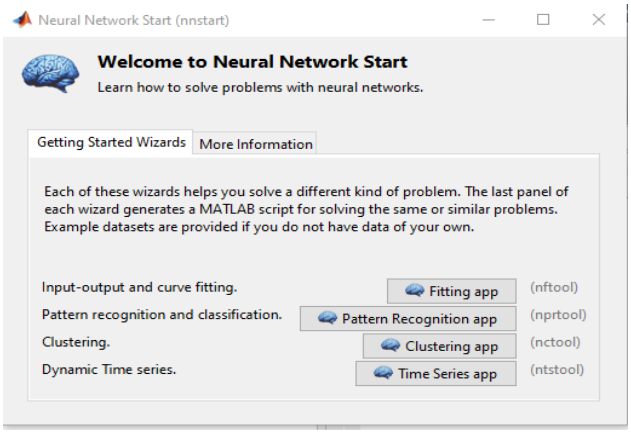
- Modified Testing Data [6 x 418]

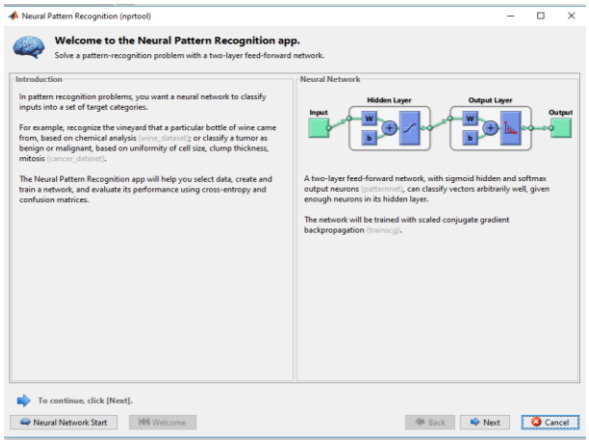
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	3	3	2	3	3	3	3	2	3	3	3	1	1
2	0	1	0	0	1	0	1	0	1	0	0	0	1
3	1	1	1	1	1	0	1	1	1	1	1	1	1
4	0	1	0	0	1	0	0	1	0	2	0	0	1
5	0	0	0	0	1	0	0	1	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	2
7													
8													

Neural Network Model

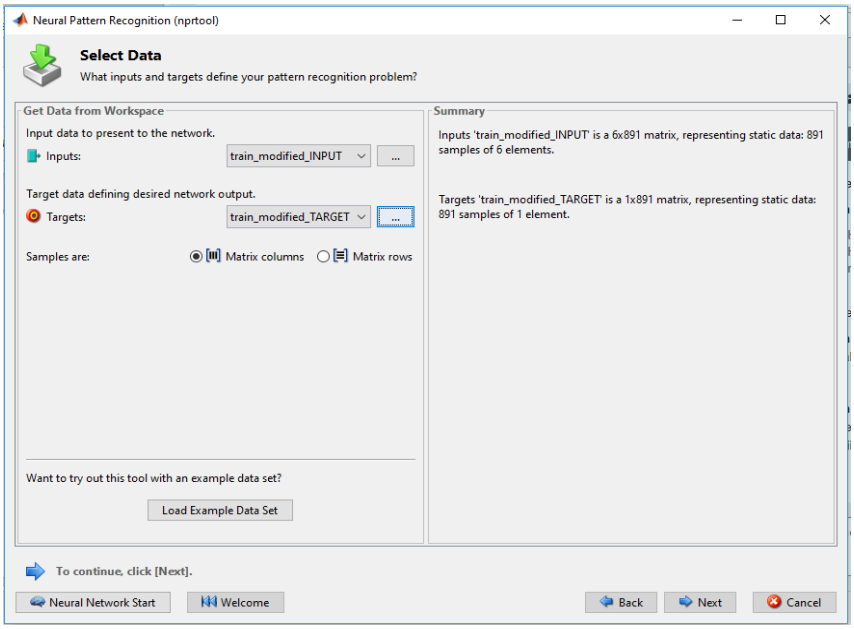
Software used - Matlab R2016a
Tools used - NN Toolbox
The model was generated by using the toolbox as follows.

01. Pattern recognition app was selected (since it is a classification problem).

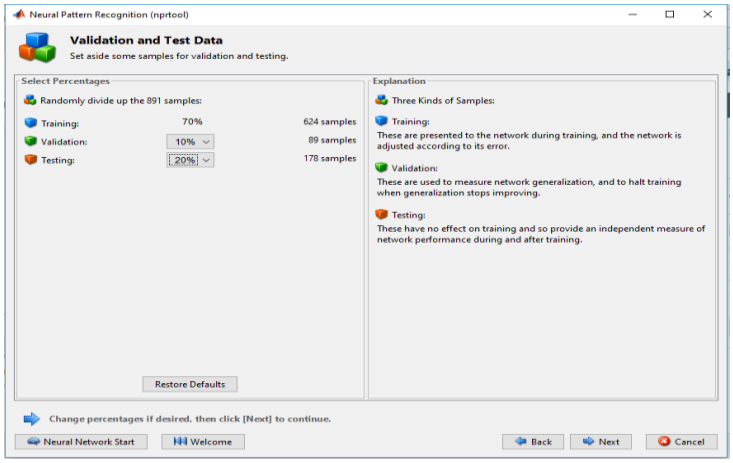




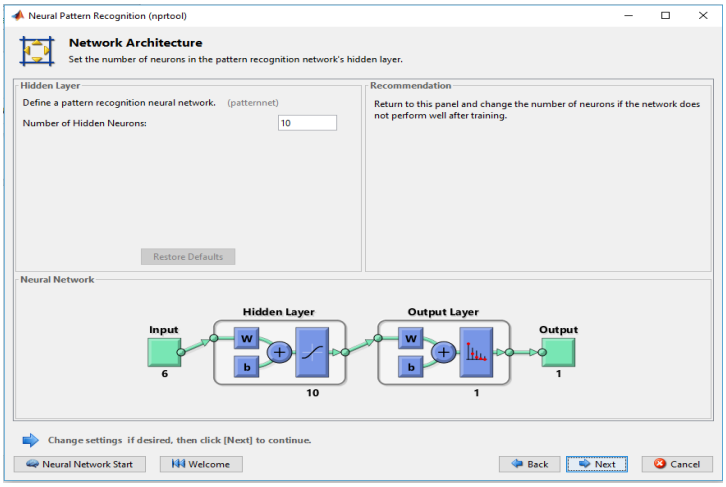
02. Modified input matrix of training data was imported as the input - 6*891
Modified target matrix of training data was imported as target - 1*891



03. Training data set was split into 3 parts - training (70%), validation (10%) and testing (20%).



04. Number of hidden nodes was selected as 10 (or as desired).



05. Training data set was trained.

Retraining and changing of number of hidden nodes were done to increase the performance.

06. Accuracy of the network was analyzed through;

- Performance plot (Cross - Entropy vs Epochs)
- Training state plot
- Error histogram (Instances vs Errors[Target - output])
- Confusion plot

NEURAL NETWORK CODE

```
%   train_modified_INPUT - input data.
%   train_modified_TARGET - target data.
%   test_modified - test data.

x = train_modified_INPUT;
t = train_modified_TARGET;
z = test_modified;

% Choose a Training Function
% 'trainscg' uses less memory. Suitable in low memory
situations.
trainFcn = 'trainscg';

% Create a Pattern Recognition Network
hiddenLayerSize = 10;
net = patternnet(hiddenLayerSize);

% Setup Division of Data for Training, Validation, Testing
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 20/100;
net.divideParam.testRatio = 10/100;

% Train the Network
[net,tr] = train(net,x,t);

% Test the training set
y = net(x);
e = gsubtract(t,y);
performance = perform(net,t,y)
tind = vec2ind(t);
yind = vec2ind(y);
percentErrors = sum(tind ~= yind)/numel(tind);

% View the Network
view(net)


% Test the trained Network using test data
```

```
Ypred = net(z);
output = Ypred(:,1:418);

% Create a matrix for the output and save the predictions in a
.csv file

f = zeros (418,1);

for i = (1:418)
     % to display the passenger number
     f(i,1) = 891+i;

     % convert the predicted probability to binary format
    f(i,2) = round(output(i));
end

csvwrite('Output.csv', f)

% Plots
% Uncomment these lines to enable various plots.
figure, plotperform(tr)
%figure, plottrainstate(tr)
%figure, ploterrhist(e)
%figure, plotconfusion(t,y)
%figure, plotroc(t,y)
```

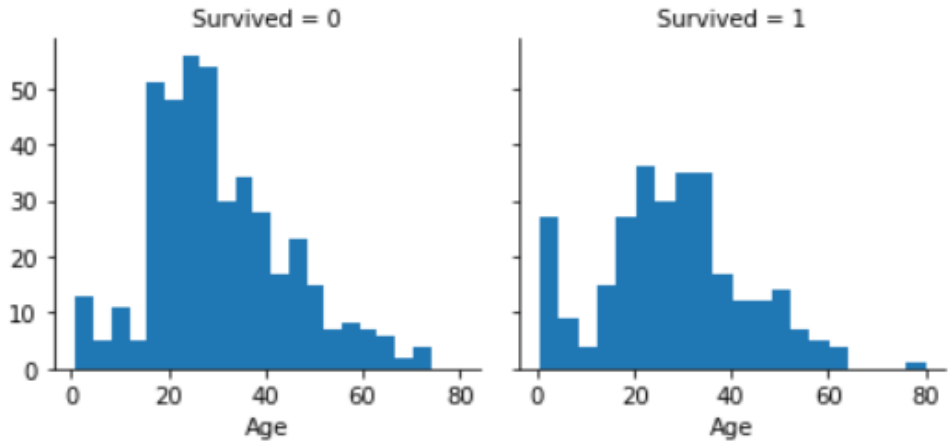
The code of the neural network was automatically generated by the software for training the network. Customized functions were encoded into the program for testing the network with test data and saving the output.

RESULTS AND ANALYSIS

Analyzing data through Visualization

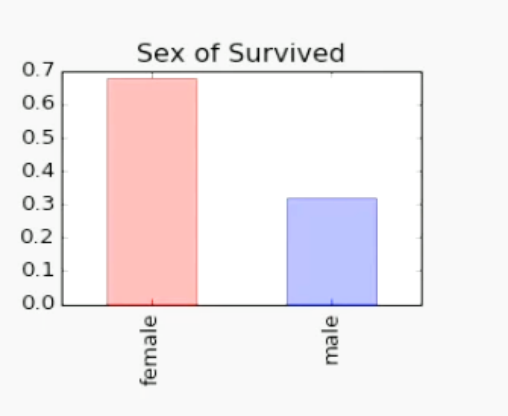
For understanding correlations between the provided numerical and categorical features and the survival rate of the passengers of titanic, separate plots of each feature with the target data of the training data were used.

01. Age



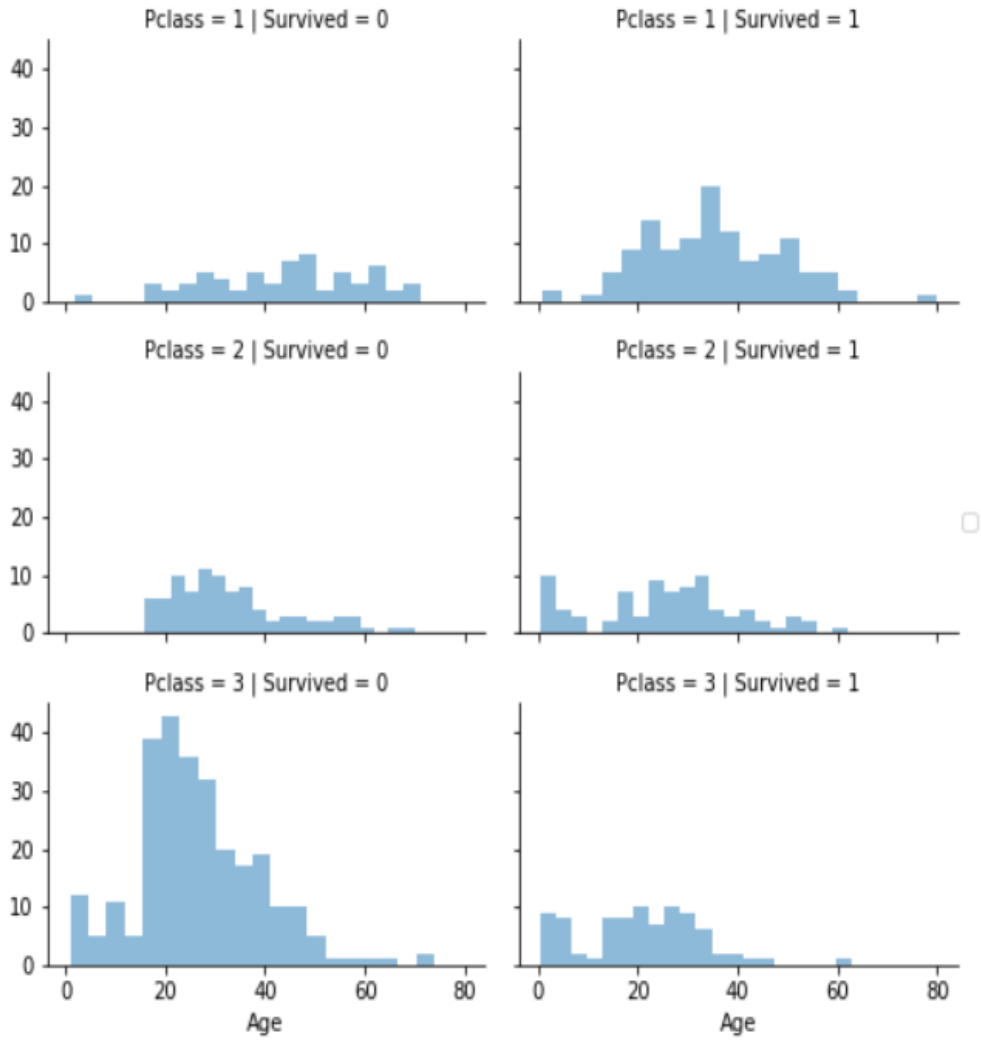
- Infants (Age <=4) had a high survival rate whereas the older passengers (Age > 60) showed very low survival rate.
- A significant proportion of the 15-25 year olds did not survive.

02. Sex



- Female passengers had a significantly higher survival rate than males.

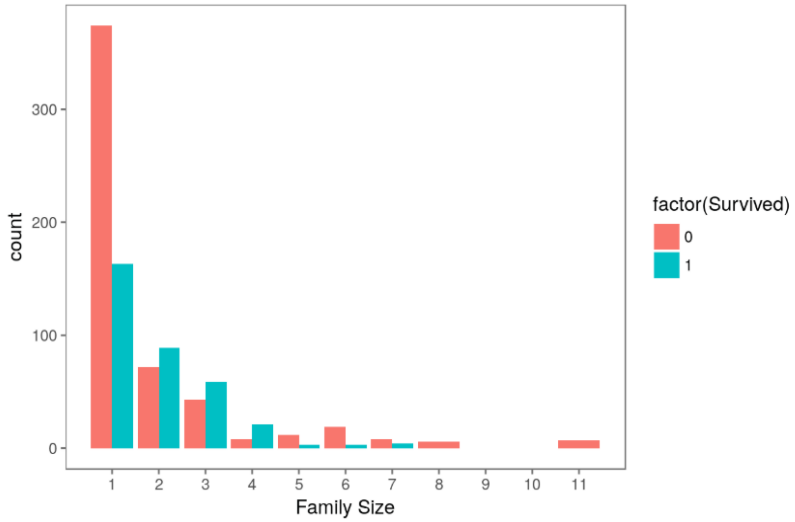
03. Class (Economic Status)



- Pclass=3 had most passengers, however most did not survive while most passengers in Pclass=1 survived.
- Furthermore, infant passengers in Pclass=2 and Pclass=3 mostly survived.
- It is also evident that pclass varies with the Age distribution of passengers.

04. Family Size (sibsp and parch)

- Family size variable based on number of siblings/spouse and number of children/parents was observed to affect the survival rate.
- Having a family increased the odds of survival, but when family size increases above 4, survival rate goes down.



- Even though the cabin feature is highly incomplete (contains many null values both in training and test dataset), it was considered since the location could factor in to the passengers' access to lifeboats.
- PassengerId was disregarded from training dataset as it does not contribute to survival.
- Name feature is relatively non-standard, and it may not contribute directly to survival.
- Embark feature was dropped from training dataset as it does not contribute to survival.

Survival Prediction

```
ans =

  Columns 1 through 12

    0.1640    0.5606    0.1828    0.1640    0.3850    0.1362    0.5632    0.1710    0.5632    0.1649    0.1640    0.3338

  Columns 13 through 24

    0.9525    0.1827    0.9456    0.9175    0.1828    0.1640    0.5606    0.5632    0.3306    0.1243    0.9552    0.2394

  Columns 25 through 36

    0.8725    0.1646    0.9435    0.1640    0.3001    0.1649    0.1827    0.1825    0.2730    0.2730    0.2479    0.1640

  Columns 37 through 48

    0.5632    0.5632    0.1640    0.1640    0.1618    0.2320    0.1640    0.9180    0.9485    0.1640    0.3001    0.1640

  Columns 49 through 60
```

The probability of survival of each passenger of the testing data set is predicted on the Command Window. These probabilities are rounded off to their respective binary values and saved onto a separate .csv file as requested by the Kaggle competition.

	A	B	C
1	892	0	
2	893	1	
3	894	0	
4	895	0	
5	896	1	
6	897	0	
7	898	1	
8	899	0	
9	900	1	
10	901	0	
11	902	0	
12	903	0	
13	904	1	
14	905	0	
15	906	1	
16	907	1	
17	908	0	
18	909	0	
19	910	1	
20	911	1	
21	912	0	

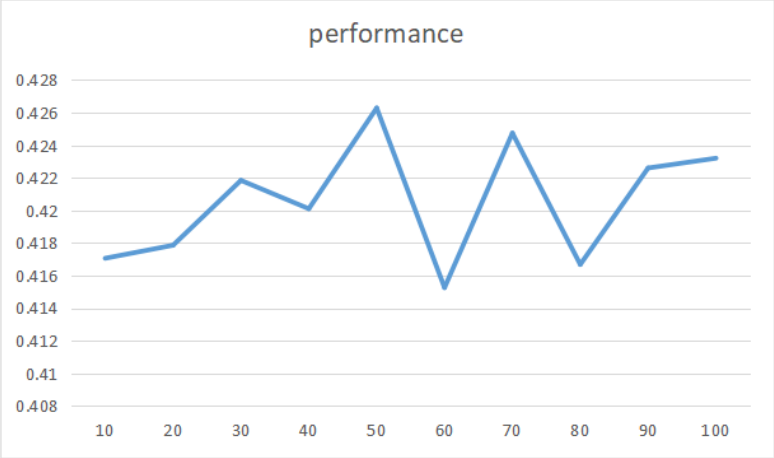
Performance

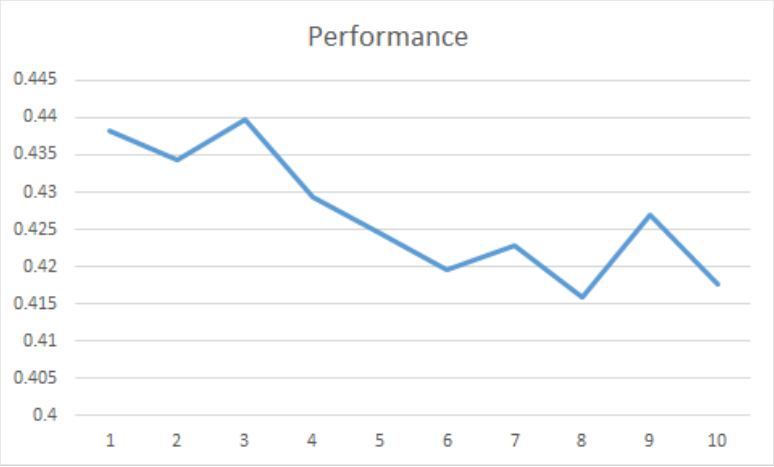
A sample output of the trained network tested with the test data provided is shown below. The probability of survival for each passenger is predicted. The network shows a performance of the range 0.42 to 0.47.

The performance of the trained network was analyzed by comparing with the output with targets. Therefore this does not an indication of the accuracy of the testing data, but the training data.

In this project, a lower performance than expected was achieved. This could be due to the small dataset provided to train the network. In order to predict accurately, the network must be trained with 1000s of data samples. However in this project, only 891 samples were provided, out of which a significant portion had missing values. This data set cannot be increased since it is representative of the real-life titanic shipwreck (new data cannot be generated).

In order to increase the performance of the neural network the data set was preprocessed and only the related features were selected. Furthermore, the variation of performance with the number of hidden nodes in the network was analyzed.



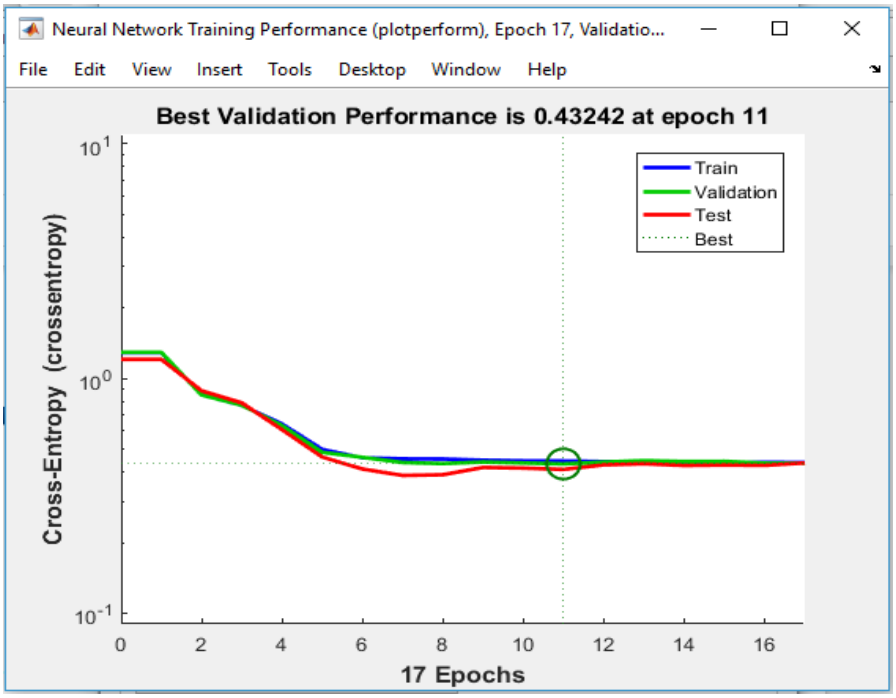


Even though a correlation exists between the number of hidden nodes and performance of the network, it is difficult to determine the specific underlying principle. By training the network with different number of hidden nodes the above graphs were obtained.

The highest average performance was observed to be present when the number of hidden nodes was 3. Having a low number of hidden nodes reduces the complexity of the network, therefore requiring less computational power in training.

Moreover, the performance of the network could be improved by tuning the algorithm and network functions through weight initialization, learning rate adjustment, activation functions, network topology, epochs, regularization and early stopping.

Cross-Entropy Plot

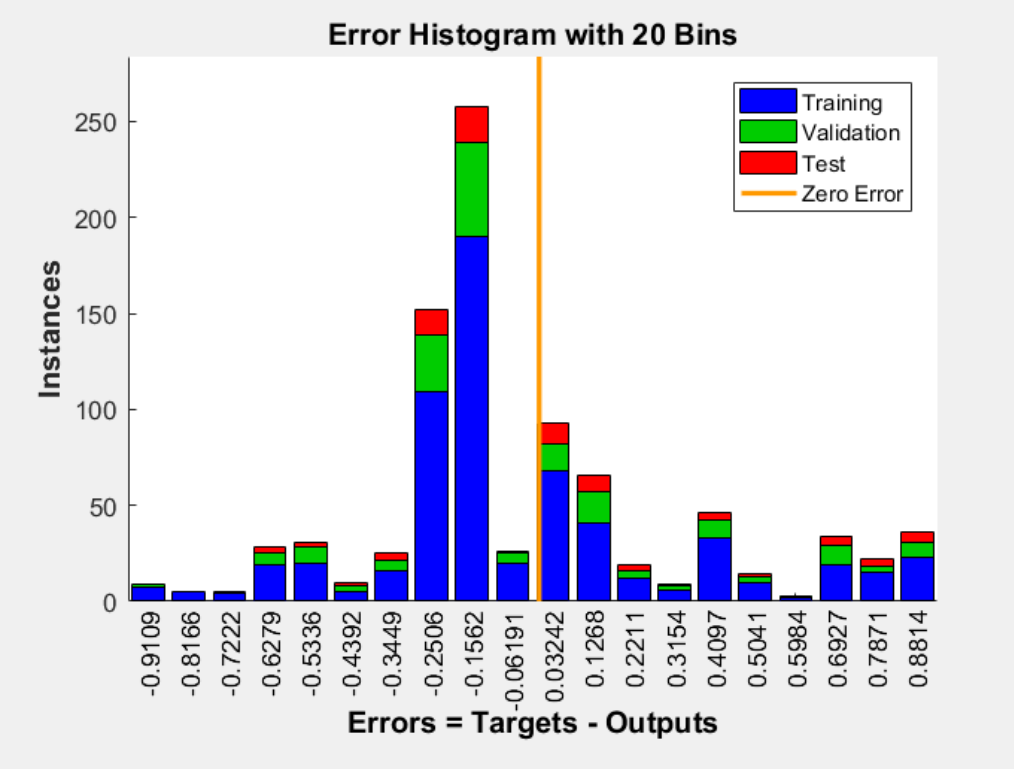


Cross entropy is used to define the loss function in machine learning and optimization. In this particular example, training had run for 13 epochs. This plot indicates the iteration at which the validation performance reached a minimum (iteration 7). The training continued for 6 more iteration before the training stopped.

Generally, the error reduces after more epochs of training, but might start to increase on the validation data set as the network starts overfitting the training data. The best performance is taken from the epoch with the lowest validation error.

This figure does not indicate any major problems with the training. The validation and test curves are very similar. If the test curve had increased significantly before the validation curve increased, then it is possible that some overfitting might have occurred.

Error Histogram



Error HIstogram shows the distribution of errors for the training, validation and test subsets. Bins are the number of vertical bars on the graph. The total error from neural network ranges from -0.9109 (leftmost bin) to 0.8814 (rightmost bin). This error range is divided into 20 smaller bins.Each vertical bar represents the number of samples from the training dataset, which lies in a particular bin, each bin has a width of

(0.8814-(-0.9109))/20 = 0.08961

For example, at the mid of the graph, the bin corresponds to the error of -0.06191 and the height of that bin for validation dataset is 25. It means that 25 samples from the validation dataset have an error lying in the following range.

(-0.06191 - 0.08961/2 , -0.06191 + 0.08961/2)

(-0.061902, 0.10671) < the range of the bin corresponding to -0.06191

It is better if the zero error lies on the highest vertical bar. When training has done many times, due to quantization of the bins, 0 (perfect match) falls into that largest bar.

Confusion plot

Confusion plot plots a confusion matrix for the true labels targets and predicted labels outputs. On the confusion matrix plot, the rows correspond to the predicted class (Output Class) and the columns correspond to the true class (Target Class). The diagonal cells correspond to observations that are correctly classified. Both the number of observations and the percentage of the total number of observations are shown in each cell.

The column on the far right and the row at the bottom of the plot show the percentages of all the examples predicted to belong to each class that are correctly and incorrectly classified. The cell in the bottom right of the plot shows the overall accuracy.

The table below shows the correctly classified data samples;

Class	Train	Validation	Test	Total
0	89.5%	85.13%	91.5%	88.9%
1	70.0%	69.6%	70.0%	69.9%

Data is classified into 2 target classes, and the training, validation and testing set performances within each class. Therefore, when class results are combined, the train, validation and test set results remain similar. Only one plot is seen on the graph because it is on top of the other two.

Classifier yielding rates of ~70.0% for class 1 should be trained more.

DISCUSSION

Data Preparation

- The given training data is used to understand the correlation between different variables and the probability of survival to classify/categorize the passengers of the test data set.
- However, not all the features provided by Kaggle contribute to the prediction of survival. These extraneous categorical and numerical variables were visualized statistically through separate plots for increased accuracy.
- Through data visualization, it was assumed that the features; ticket number, point of embarkation, name and fare will not have a direct influence on survival.
- After sorting the data, the data should be prepared to be compatible with the neural network model. For this all features (including categorical) were converted to numerical values (discrete values). In discrete data, it is easier to identify a pattern than continuous data. Therefore it is easily integrated into many different types of statistical and mathematical functions of neural networks.
- Missing data is an issue in the datasets provided and this may have a significant effect on the conclusions that drawn from the data. The records with missing values cannot be simply removed because of the size of the dataset is already small and such action would reduce the data available for training and testing the network. Ideally the missing values should be estimated to ensure the accuracy of the network, but in this model it is difficult to estimate values since they are categorical and real life data and a small deviation could impact the prediction significantly.
- Classification is done considering the assumptions based on the problem.
  - ✓ Women (Sex=female) were more likely to have survived.
  - ✓ Children (Age<18) were more likely to have survived.
  - ✓ The upper-class passengers (pclass=1) were more likely to have survived.
- The data matrix was transposed to a column matrix since it is the standard input data form of neural networks.

Model Selection

The type of the problem and solution requirement should be understood clearly in order to select the best predictive modeling algorithm.

The relationship between variables and the survival rate is identified and new data is categorized. Therefore, the titanic problem is a classification and regression problem employing supervised machine learning techniques.

With these two criteria, Supervised Learning, and Classification and Regression, the possible network models are;

- Logistic Regression
- KNN or k-Nearest Neighbors
- Support Vector Machines
- Naive Bayes classifier
- Decision Tree
- Random Forest
- Perceptron
- Artificial neural network
- RVM or Relevance Vector Machine

Perceptron cannot be used since it is a simple network with no hidden layer, so the accuracy is low. Logistic regression is ideal for this purpose since it is a categorical classification but it can lead to overfitting of data and variables should be independent of each other. K-Nearest Neighbor (KNN) Classifier is a simple classifier that works well on basic recognition problems but not robust to noisy data. Also it does not learn from the training data set, it simply uses the training data itself for classification based on distances.

In this project, the Artificial Neural Network model was used.

The artificial neural network will speed up the computations. The neural network will converge at the local minima and not the global minima in the loss function. This can be overcome by optimizing the model, with increasing the learning rate slightly. Another strategy employed is the learning rate decay, which reduces the learning rate every epoch. This can also prevent from getting stuck at a local minima in the training phase and can achieve a good result at the end. Even though neural networks take more computational time, the prediction is highly accurate.

Data analysis

Data matrix

Input matrix is a 981*5 matrix. 981 corresponds to the number of people in the training set and 6 corresponds to the features (sex, age, sibsp, parch, pclass, cabin) which have an influence. This was split as 70% of the original train data for training, and 10% of it for validation and 20% of it for testing. Test set is used to prevent data from overfitting. If overfitting happens with the training data, model will not classify test data (data given without labels) correctly.

Hidden layer

Hidden layer refers to the number of neurons that should be present in the hidden layer to obtain the optimal accuracy (performance). The maximum accuracy was achieved when using 3 neurons and the performance decreases when we increase the number of neurons. This may be due to number of training sample being low.

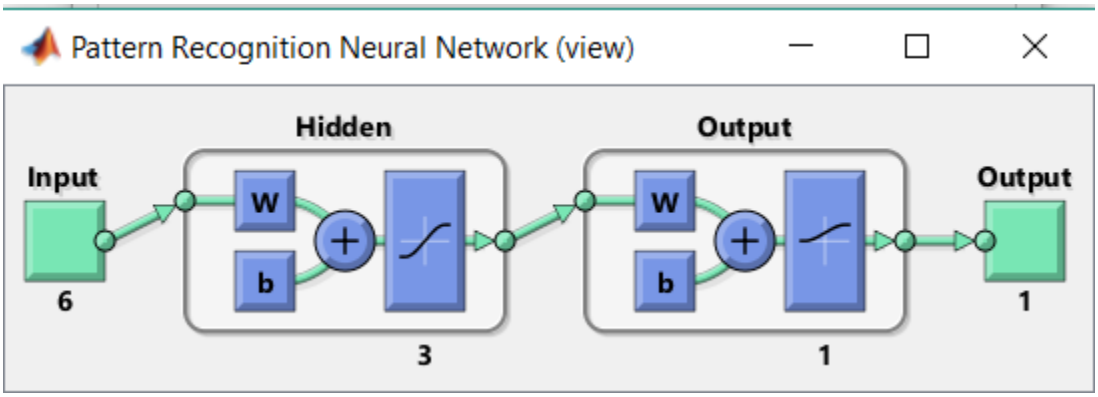
Training Function

Trainscg function was used to train the network. trainscg can train any network as long as its weight, net input, and transfer functions have derivative functions. Backpropagation is used to calculate derivatives of performance with respect to the weight and bias variables. It updates weight and bias values according to the scaled conjugate gradient backpropagation(SCG). SCG is a supervised learning algorithm for feedforward neural networks. The basic backpropagation algorithm adjusts the weights in the steepest descent direction. In the conjugate gradient algorithms a search is performed using second order techniques along conjugate directions, which produces generally faster convergence than steepest descent directions.'trainscg' uses less memory. Therefore, it is suitable in low memory situations.

Training stops when any of these conditions occurs:

- ✓ The maximum number of epochs (repetitions) is reached.
- ✓ The maximum amount of time is exceeded.
- ✓ Performance is minimized to the goal.
- ✓ The performance gradient falls below min_grad.
- ✓ Validation performance has increased more than max_fail times since the last time it decreased (when using validation).

Neural network



- Input nodes - 6
- Output nodes - 1
- Hidden layer – 1 (with 3 neurons)
- Weights and biases are added at input and hidden layers

Input layer weight matrix:

iw | iw{1, 1} | lw | lw{2, 1}

iw{1, 1}

1	2	3	4	5	6
-0.0924	0.4189	-0.1765	0.9230	0.9379	-0.8327
-1.2160	0.9703	-0.3613	0.0332	-0.0546	0.3214
0.4663	0.5451	-1.1510	-0.4513	0.6580	0.0838

This is a 3*6 weight matrix. Ffor all 6 input nodes in the input layer, weights are added. Then it is provided as the input for the 3 hidden nodes in the hidden layer.

Hidden layer weight matrix:

iw | iw{1, 1} | lw | lw{2, 1}

lw{2, 1}

1	2	3	4
0.2313	2.3972	1.1270	

This is a 1*3 weight matrix. For all 3 hidden nodes in the hidden layer, weights are added. Then it is provided as the input for the output node in output layer.

CONCLUSION

Visualization and Analyzing Data

Visualization is an important statistical tool in identifying the hidden patterns among the features. For obtaining a good understanding of the dataset, it is required to analyze data and identify the relationships among features. Such patterns will give an insight into the data set which is helpful in forming the neural network and for preprocessing data.

Preprocessing

Preprocessing of data is a crucial step given the nature of the provided dataset. Transformation and normalization of data, formation of matrices and filling missing values were done to maintain the integrity of data while increasing accuracy.

Neural Network

An appropriate machine learning model should be selected based on the data set. Otherwise, precious time and resources will be wasted on trying to model the dataset with an incorrect tool.

Training

The survival rate of passengers was predicted using the trained neural network in a binary format.

Analysis

Even though the predictions about the survival were made from the trained network, the performance was low. When analyzing the dataset, that the training data set is small and some data were missing. Even though there are statistical mechanisms to fill those missing values, they cannot be applied in this study. If the data set was large, the model may have predicted more accurately.

Contribution

Preprocessing of data, building of the neural network model and making the report was done as a joint effort of all three group members.