Compare Support Vector Machines to a 3 Layer Neural Networks on Titanic Dataset

PLABANJIT KARMAKAR

Machine Learning Engineer Intern
at
AI Technology & Systems
https://www.ai-techsystems.com
Kolkata, India
plabanjit.karmakar22@gmail.com

Abstract **-** The luxury steamship RMS Titanic created quite a stir when it departed for its maiden voyage from Southampton, England, on April 10, 1912. Titanic sank in the early hours of April 15, 1912, off the coast of Newfoundland in the North Atlantic after sideswiping an iceberg during its maiden voyage which made it one of the deadliest peacetime maritime disasters in history. Of the 2,240 passengers and crew on board, more than 1,500 lost their lives in the disaster. Titanic has inspired countless books, articles and films, and her story has entered the public consciousness as a cautionary tale about the perils of human hubris. This fateful & mysterious incident still pushes the researchers & analysts to study different aspects of it. Here, we will go through the process of creating two different Machine Learning models on the famous Titanic dataset and compare them. It will give some idea on the fate of passengers on the Titanic, summarized according to economic status (class), sex, age and survival etc. [1]

Keywords – Support Vector Machines (SVM), Neural Network, Prediction, Classification, Confusion Matrix, Accuracy, R, Machine Learning.

I. INTRODUCTION

On April 14, 1912, the R.M.S. Titanic collided with a massive iceberg and sank in less than three hours. At the time, more than 2200 passengers and crew were aboard the Titanic for her maiden voyage to the United States. Only 705 survived. Even after so many years there are so many rumors and speculations regarding the actual cause and survival rate of the passengers in this unfortunate disaster. Here we will apply Machine Learning (ML) algorithms to explore the insights of this incident based on the available data. Over the years the data on survived and deceased passengers are collected. This data is available on Kaggle.com. [2] ML is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to perform the task. [3]

Here our key objective is to apply two different ML algorithms named SVM (Support Vector Machines) and MLP Neural Network (Multi-layer Perceptron Neural Network) and compare their accuracy in percentage on the dataset along with exploring various aspects and characteristics of the dataset.

II. DATASET

The dataset we use for our paper was provided by the Kaggle.com website. It is a free open source platform to all. The data has been split into two groups: training set (train.csv) and test set (test.csv). **The training set** should be used to build your machine learning models. **The test set** should be used to see how well your model performs on unseen data. The data consists of 891 rows in the train set which is a passenger sample with their associated labels. For each passenger, we were also provided with the name of the passenger, sex, age, his or her passenger class, number of siblings or spouse on board, number of parents or children aboard, cabin, ticket number, fare of the ticket and embarkation. The data is in the form of a CSV (Comma Separated Value) file. For the test data, we were given a sample of 418 passengers in the same CSV format. The Description of Dataset fields in titanic is listed below: [4]

Description of Dataset fields in titanic

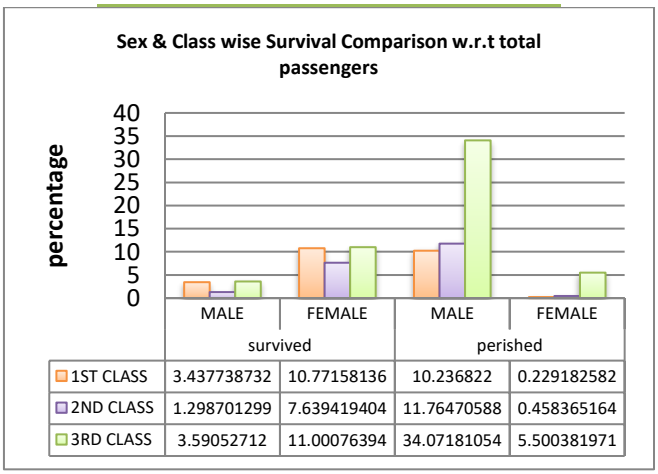
Passenger ID - Identification no. of the Passengers
survival - Survival (0 = No; 1 = Yes)
class - Passenger Class (1 = 1st; 2 = 2nd; 3 = 3rd)
name - Name

sex - Sex
age - Age
sibsp - Number of Siblings/Spouses Aboard
parch - Number of Parents/Children Aboard
ticket - Ticket Number
fare - Passenger Fare
cabin - Cabin
embarked - Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)

III. DATA INSIGHTS

Table 1 : SNAPSHOT

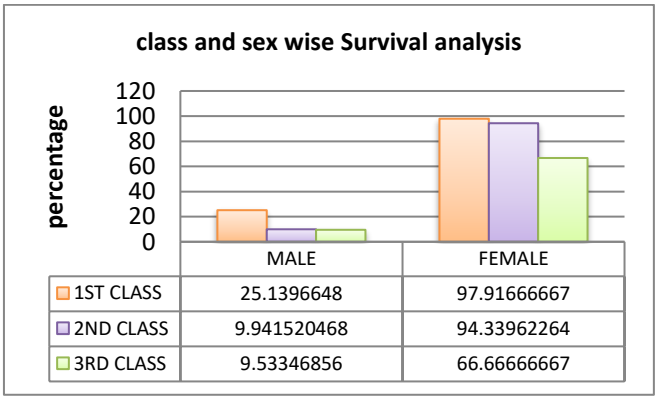
% of total passengers	Sex	Survival Status
8.32696715	MALE	SURVIVED
29.41176471	FEMALE	SURVIVED
56.07333843	MALE	PERISHED
6.187929717	FEMALE	PERISHED



	survived MALE	survived FEMALE	perished MALE	perished FEMALE
1ST CLASS	3.437738732	10.77158136	10.236822	0.229182582
2ND CLASS	1.298701299	7.639419404	11.76470588	0.458365164
3RD CLASS	3.59052712	11.00076394	34.07181054	5.500381971

Chart 2: Sex & Class wise Survival Comparison w.r.t total passengers

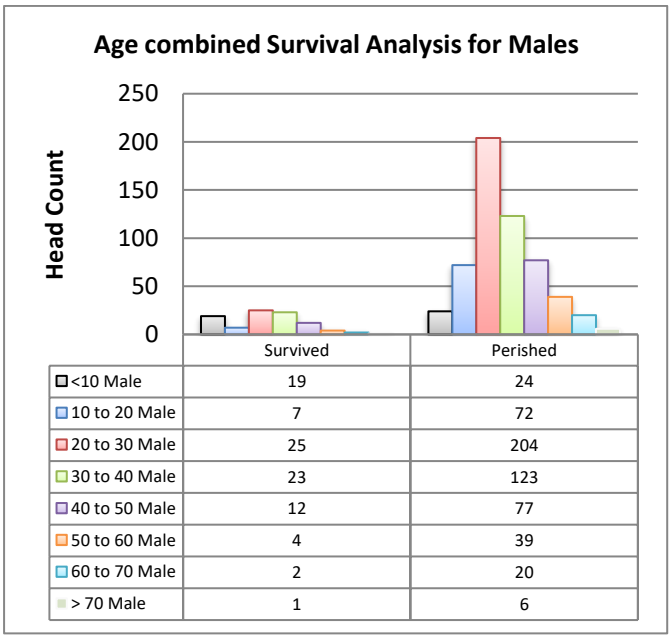
We know that women and children were given preference in the rescue process. From the dataset it is found that 56% of total passengers were males who perished whereas 6% of total passengers were females who perished in this disaster. We also find that most of the 3rd class passengers contribute (around 54%) in the Perished Category.



	MALE	FEMALE
1ST CLASS	25.1396648	97.91666667
2ND CLASS	9.941520468	94.33962264
3RD CLASS	9.53346856	66.66666667

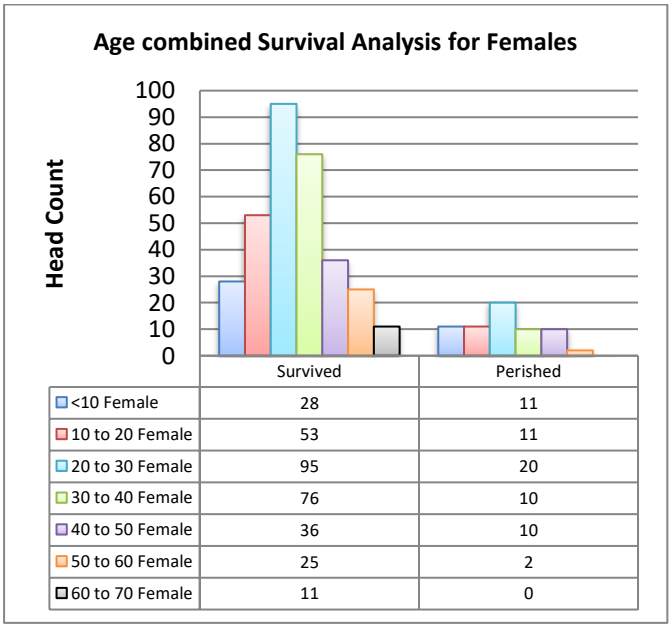
Chart 2: sex wise Survival analysis w.r.t each class

This chart shows that 25% of the 1st class Male passengers managed to avoid death, whereas 2nd and 3rd class male passengers were not lucky enough, the same argument can be applied on the 3rd class Female passengers also. There is no doubt about the fact that even in a sinking ship, Money talks.



	Survived	Perished
<10 Male	19	24
10 to 20 Male	7	72
20 to 30 Male	25	204
30 to 40 Male	23	123
40 to 50 Male	12	77
50 to 60 Male	4	39
60 to 70 Male	2	20
> 70 Male	1	6

Chart 3: Age combined Survival Analysis for Males



	Survived	Perished
<10 Female	28	11
10 to 20 Female	53	11
20 to 30 Female	95	20
30 to 40 Female	76	10
40 to 50 Female	36	10
50 to 60 Female	25	2
60 to 70 Female	11	0

Chart 4: Age combined Survival Analysis for Females

These charts have been prepared excluding the missing age from the raw data.
Though the children were the 1st priority in the rescue process, but many of them could not make it at the end in that adverse situation. Most of the males between age 20 to 40 perished in that disaster.
In case of females due to the Birkenhead Drill, the scenario is comparatively better. Data shows that most of the females were rescued.

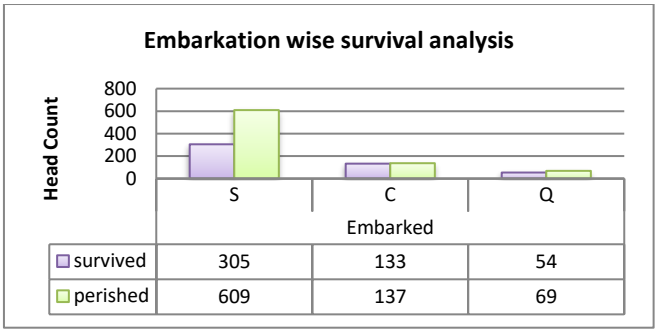


Chart 5: Embarkation wise survival analysis

This chart depicts the fact that most of the passengers embarked from Southampton and the survival rate of the passengers from this port is lower compared to other Embarkation ports.

Now let us study the following charts prepared using matplotlib and seaborn package in python including the missing values. [13]

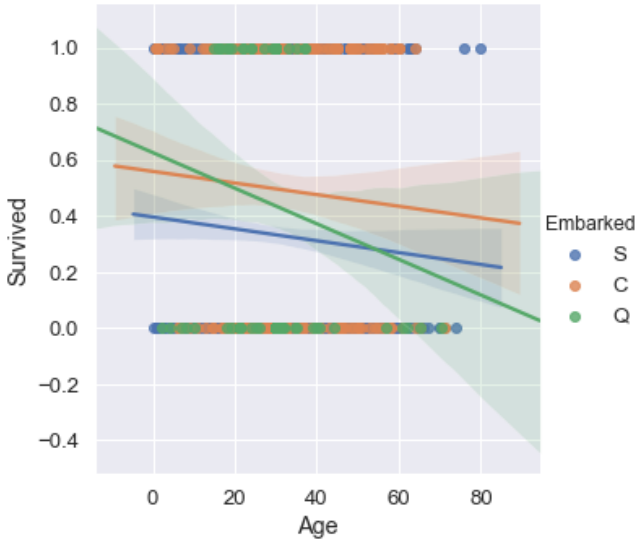


Chart 6: Embarkation & age combined survival analysis

On an average the survival rate of passengers decreaces as the age increases, irrespective of Embarkation ports.

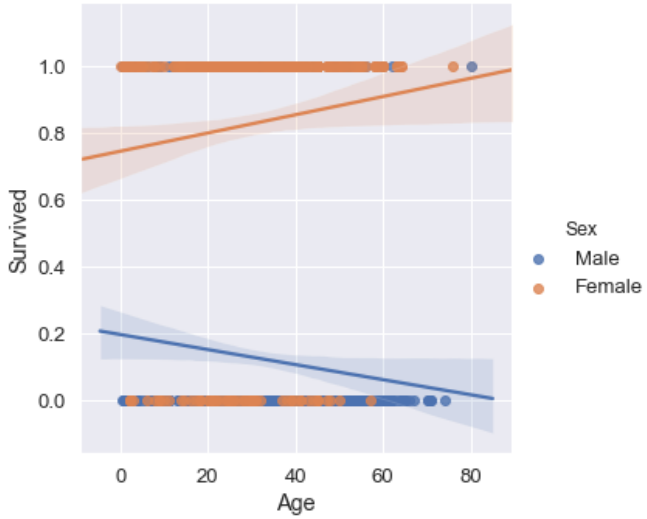


Chart 7: Sex & age combined survival analysis

If we observe the above graph carefully, we can see that on an average male survival rate is very low and it decreases simultaneously as the age increases. But in case of women the survival rate is very high compared to males and the survival rate has not come down as the age increases. Clearly these two combined factors classify the data.

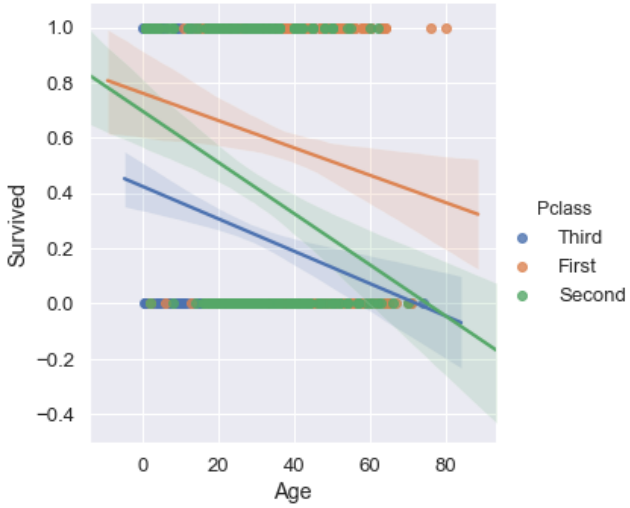


Chart 8: Passenger Class & age combined survival analysis

Again, here we can see that on an average the survival rate decreases as age increases. But there is a constant gap in the survival rate between 1st class and 3rd class passengers. In case of 2nd class passengers the rate is moderately high in the early ages but as the age increases survival rate decrease rapidly.

IV. THE R ENVIRONMENT

Here we are using R language for our analysis because we have achieved excellent accuracy using SVM in this Language. R is a language and environment for statistical computing and graphics. R provides a wide variety of statistical (linear and nonlinear modeiling, classical statistical tests, time-series analysis, classification, clustering …) and graphical techniques, and is highly extensible. [5]

R is one of the major languages for data science. It provides excellent visualization features, which is essential to explore the data before submitting it to any automated learning, as well as assessing the results of the learning algorithm. Many R packages for machine learning are available off the shelf and many modern methods in statistical learning are implemented in R as part of their development. [6]

V. DATA PREPROCESSING

For machine learning algorithms to work it is necessary to convert the raw data into a clean data set and dataset must be converted to numeric data. You have to encode

all the categorical labels to column vectors with binary values. Missing values or NaNs in the dataset is an annoying problem. You have to either drop the missing rows or fill them up with a mean or interpolated values. Both the training data and test data must have same dimensions for the model. [7]

A. Dropping Columns which are not useful

At first step we will drop some of the columns which many not contribute much to our machine learning model such as Name, Passenger ID and Cabin.

B. Dealing with Missing values

We see that the feature Age has 177 missing values in training data and 86 missing values in test data. Dropping all these rows means we are wasting data**.** Machine learning models need data for training to perform well. So we take the mean age of the rest ages and fill the blank places with it. After this we drop all the rows with missing values in both training and test data.

At this stage first 6 rows of our training and test data looks like this –

```
> head(train)
  Survived Pclass    Sex      Age SibSp Parch    Fare Embarked
1        0      3   male 22.00000     1     0  7.2500        S
2        1      1 female 38.00000     1     0 71.2833        C
3        1      3 female 26.00000     0     0  7.9250        S
4        1      1 female 35.00000     1     0 53.1000        S
5        0      3   male 35.00000     0     0  8.0500        S
6        0      3   male 29.69912     0     0  8.4583        Q
> head(test)
  Survived Pclass    Sex  Age SibSp Parch    Fare Embarked
1        0      3   male 34.5     0     0  7.8292        Q
2        1      3 female 47.0     1     0  7.0000        S
3        0      2   male 62.0     0     0  9.6875        Q
4        0      3   male 27.0     0     0  8.6625        S
5        1      3 female 22.0     1     1 12.2875        S
6        0      3   male 14.0     0     0  9.2250        S
```

Table 2: Head of Training and test data in SVM

In case of MLP NN we have to convert the characters into numerical data. We simply assign numbers to each category. At this stage first 6 rows of our training and test data looks like this -

```
> head(train)
  Survived Pclass Sex      Age SibSp Parch    Fare Embarked
1        0      3   0 22.00000     1     0  7.2500        3
2        1      1   1 38.00000     1     0 71.2833        1
3        1      3   1 26.00000     0     0  7.9250        3
4        1      1   1 35.00000     1     0 53.1000        3
5        0      3   0 35.00000     0     0  8.0500        3
6        0      3   0 29.69912     0     0  8.4583        2
> head(test)
  Survived Pclass Sex  Age SibSp Parch    Fare Embarked
1        0      3   0 34.5     0     0  7.8292        2
2        1      3   1 47.0     1     0  7.0000        3
3        0      2   0 62.0     0     0  9.6875        2
4        0      3   0 27.0     0     0  8.6625        3
5        1      3   1 22.0     1     1 12.2875        3
6        0      3   0 14.0     0     0  9.2250        3
```

Table 3: Head of Training and test data in MLP NN

VI. ALGORITHMS USED IN ANALYSIS

A. Support vector machine (SVM):

It is a supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. It is mostly used in classification problems. In this algorithm, each data item is plotted as a point in n-dimensional space (where n is number of features), with the value of each feature being the value of a particular coordinate. Then, classification is performed by finding the hyper-plane that best differentiates the two classes.

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification, implicitly mapping their inputs into high-dimensional feature spaces. A SVM is a discriminative classifier formally defined by a separating hyper plane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyper plane which categorizes new examples. [8]

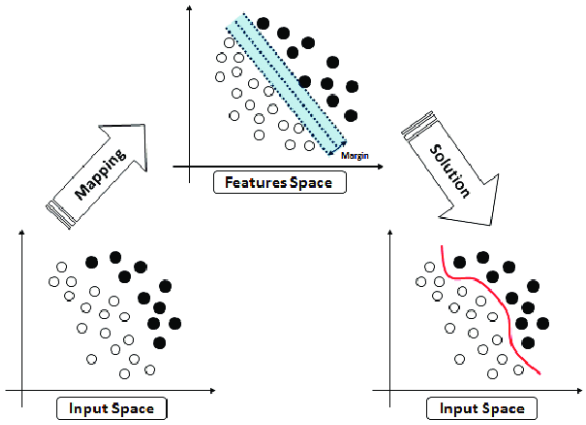


Fig 1: SVM process flow for Classification Problem

Here we are using kernlab package in R to run our SVM Algorithm. Kernlab is an extensible package for kernel-based machine learning methods in R. The package contains implementations of support vector machines and the relevance vector machine. [9]

B. MLP Neural Network (or Artificial Neural Network):

It has the ability to learn by examples. ANN is an information processing model inspired by the biological neuron system. It is composed of a large number of highly interconnected processing elements known as the neuron to solve problems. It follows the non-linear path and process information in parallel throughout the nodes. A neural network is a complex adaptive system.

Adaptive means it has the ability to change its internal structure by adjusting weights of inputs. [10]

A neural network consists of:

1. *Input layers:* Layers that take inputs based on existing data. Here we have 7 input variables.
2. *Hidden layers:* Layers that use back propagation to optimize the weights of the input variables in order to improve the predictive power of the model. Here we are using 1 hidden layer with 4 hidden units (neuron).
3. *Output layers:* Output of predictions based on the data from the input and hidden layers.

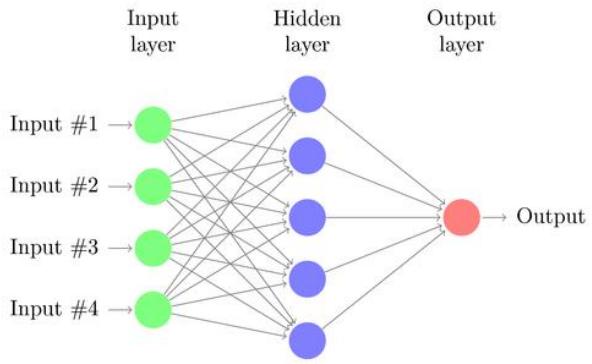


Fig 2: 3 layer MLP NN with 5 neurons in the hidden layer.

In this particular example, our goal is to develop a neural network to determine if an individual will survive or not. Here, we are using the neural network to solve a classification problem. By classification, we mean ones where the data is classified by categories. In our dataset, we assign a value of **1** to an individual survived and value of **0** to an individual who could not survive.

Our independent variables are as follows:

Pclass, Sex, Age, SibSp, Parch, Fare and Embarked

Data Normalization

One of the most important procedures when forming a neural network is data normalization. This involves adjusting the data to a common scale so as to accurately compare predicted and actual values. Failure to normalize the data will typically result in the prediction value remaining the same across all observations, regardless of the input values.

We can do this in two ways in R:

1. Scale the data frame automatically using the *scale* function in R

2. Transform the data using a *max-min normalization* technique. Here we are using this technique. [11]

Here we have used the neuralnet package to build NN classifier model. The package allows flexible settings through custom-choice of error and activation function. Our NN Model plot looks like this.

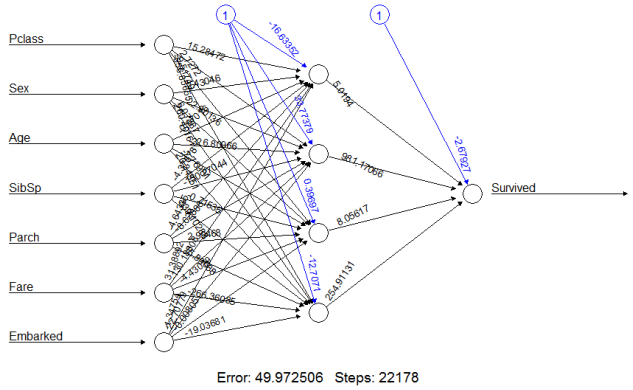


Fig 3: 3 layer MLP NN with 4 neurons in the hidden layer.

VII. PERFORMACE OF THE ALGORITHMS

To compare the performance of these two algorithms we will use Confusion matrix and then we will evaluate the accuracy of the model on both training data and test data.

A. Confusion Matrix: A confusion matrix is a summary of prediction results on a classification problem.

The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix. The confusion matrix shows the ways in which your classification model is confused when it makes predictions. It gives us insight not only into the errors being made by a classifier but more importantly the types of errors that are being made.

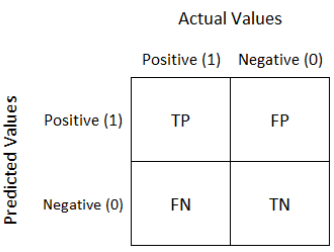


Fig 4: Confusion Matrix

Definition of the Terms:

True Positive (TP): Observation is positive, and is predicted to be positive.

False Negative (FN): Observation is positive, but is predicted negative.

True Negative (TN): Observation is negative, and is predicted to be negative.

False Positive (FP): Observation is negative, but is predicted positive.

B. Classification Rate/Accuracy:

Classification Rate or Accuracy is given by the relation: [12]

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

$$Accuracy = \frac{(\,No.of\ Correct\ Predictions\,)}{Total\ no\ of\ predictions\ made}$$

VIII. RESULTS & CONCLUSION

Observed Confusion Matrix on both algorithms:

Table 4 : CONFUSION MATRIX IN MLP NN

predicted_test					predicted_train			
	0	1	Total			0	1	Total
0	246	25	271		0	510	95	605
1	19	127	146		1	39	247	286
Total	265	152	417		Total	549	342	891

Table 5 : CONFUSION MATRIX IN SVM

predicted_test					predicted_train			
	0	1	Total			0	1	Total
0	265	6	271		0	489	116	605
1	0	146	146		1	60	226	286
Total	265	152	417		Total	549	342	891

Observed Accuracy on both algorithms:

Table 6 : ACCURACY IN PERCENTAGE

	MLP NN	SVM
Train Data	84.96072	80.24691
Test Data	89.44844	98.56115

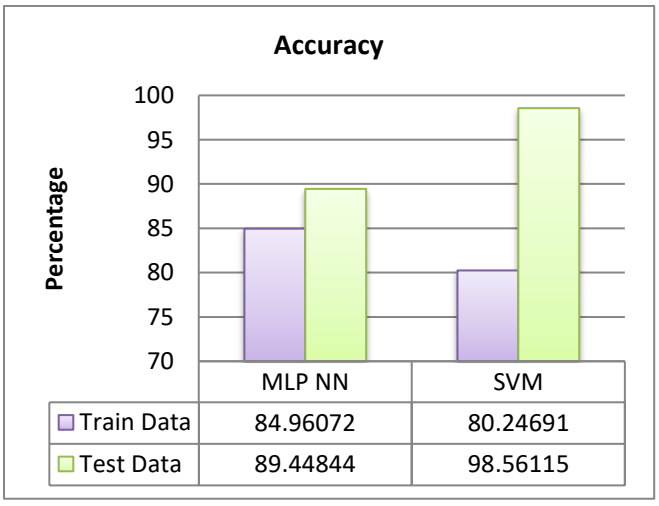


Chart 9: Accuracy Comparison of the models

From this chart we observe that in case of training data we get moderately same accuracy in both the algorithms. But in case of testing data SVM is much better than MLP NN in terms of accuracy. It is obvious that SVM is better for predictive modelling compared to MLP NN.

REFERENCES

[1] https://www.history.com/topics/early-20th-century-us/titanic#section_3
[2]https://www.kaggle.com/c/titanic/overview
[3] https://github.com/ai-techsystems/internShowcase/blob/master/prashanth.v945%40gmail.com/Third_AITS_assignment/titanic%20report%20.pdf
[4] https://github.com/awesomedata/awesome-public-datasets/issues/351
[5] https://www.r-project.org/about.html
[6] https://lgatto.github.io/IntroMachineLearningWithR/an-introduction-to-machine-learning-with-r.html
[7] https://towardsdatascience.com/implementation-of-data-preprocessing-on-titanic-dataset-6c553bef0bc6
[8] https://www.geeksforgeeks.org/classifying-data-using-support-vector-machinessvms-in-r/
[9] https://cran.r-project.org/web/packages/kernlab/vignettes/kernlab.pdf
[10]https://www.datacamp.com/community/tutorials/neural-network-models-r
[11] https://datascienceplus.com/neuralnet-train-and-test-neural-networks-using-r/
[12] https://www.geeksforgeeks.org/confusion-matrix-machine-learning/
[13] https://www.kaggle.com/ravaliraj/titanic-data-visualization-and-ml