Name – Manish kumar

Enrollment – 2020ITB007

[**Machine Learning Lab**](https://classroom.google.com/c/NTQxMjY5MTE3NDkx)

**1: Load the Titanic dataset and split it into train and test sets in an 8:1 ratio:**

import seaborn as sns

titanic=sns.load\_dataset('titanic')

titanic

from sklearn.model\_selection import train\_test\_split

Y=titanic.iloc[0:,0]

X=titanic.iloc[0:,1:13]

Y, X

**2: Divide the datasets into features (XS) and targets (YS):**

# Define the list of features to use

features = ['Pclass', 'Sex']

# Create the feature matrices for train and test sets

X\_train = train[features].values

X\_test = test[features].values

# Create the target vectors for train and test sets

y\_train = train['Survived'].values

y\_test = test['Survived'].values

X\_train,X\_test, y\_train, y\_test

**3: Create a Keras model to predict survival from the features:**

import keras

from keras import Model, Input

from keras.layers import Dense

inp = Input(shape=(9,))

l1 = Dense(6, activation='sigmoid')(inp)

out = Dense(1, activation='sigmoid')(l1)

model = Model(inputs=inp, outputs=out)

model.summary()

**4: Train the model on the training dataset:**

# Train the model

model.fit(X\_train, y\_train, epochs=50, batch\_size=32, verbose=1)

**5: Find the training and testing accuracy:**

my\_accuracy=[]

a=0

for i in range(x\_test.shape[0]):

a=model.predict(x\_test)[i]

if ( (a>=0.5 and y\_test[i]) or (a<0.5 and not(y\_test[i]) ) ) :

my\_accuracy.append(1)

else:

my\_accuracy.append(0)

my\_accuracy

**6: Check for overfitting:**

# Calculate the difference between the training accuracy and the testing accuracy

cnt=0

for i in range(len(my\_accuracy)):

cnt+=my\_accuracy[i]

print(cnt/len(my\_accuracy)\*100,"%")

**Accuracy – 87.64%**