

# CS 6375

## ASSIGNMENT 2

Names of students in your group:

Vaibhav Tyagi - VXT200018

Roshni Johnson Nambiaparambil - RXN200022

Number of free late days used: 0

Note: You are allowed a total of 4 free late days for the entire semester. You can use at most 2 for each assignment. After that, there will be a penalty of 10% for each late day.

Please list clearly all the sources/references that you have used in this assignment.

## 2. Programming Part Report

From the UCI ML repository, the Wine Quality dataset was chosen to train and test the neural network. The dataset can be found here - <https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-white.csv>

The Input variables are:

- 0 - fixed acidity
- 1 - volatile acidity
- 2 - citric acid
- 3 - residual sugar
- 4 - chlorides
- 5 - free sulfur dioxide
- 6 - total sulfur dioxide
- 7 - density
- 8 - pH
- 9 - sulphates
- 10 - alcohol
- 11 - quality

The following parameters were tuned in order to attain better results:

- Learning Rate
- Max Iterations
- Training-Testing Data%

Output for the dataset summarized in a tabular format for different combination of parameters are listed in the tables below:

## 1. Activation Function= Sigmoid

Learning rate	Number of Hidden Layers	Max iterations	Accuracy (%)
0.01	2	100	50.64
0.01	2	200	51.95
0.01	3	100	48.46
0.01	3	200	50.97
0.1	2	100	37.30
0.1	2	200	35.53
0.1	3	100	39.01
0.1	3	200	37.87

## 2. Activation Function= Tanh

Learning rate	Number of Hidden Layers	Max iterations	Accuracy (%)
0.01	2	100	41.60
0.01	2	200	41.74
0.01	3	100	39.42
0.01	3	200	40.68
0.1	2	100	35.23
0.1	2	200	36.26
0.1	3	100	34.69
0.1	3	200	35.75

### 3. Activation Function= ReLu

Learning rate	Number of Hidden Layers	Max iterations	Accuracy (%)
0.01	2	100	51.27
0.01	2	200	52.14
0.01	3	100	52.06
0.01	3	200	51.35
0.1	2	100	45.36
0.1	2	200	45.36
0.1	3	100	45.36
0.1	3	200	45.36

It is observed from the above tables and the graphs that ReLu output the better accuracy, followed by Sigmoid. ReLu is better because it is simple, fast, and empirically it seems to work well. Also training a deep network with ReLu tended to converge much more quickly and reliably than training a deep network with other activation functions.