**Aim:**

To construct a deep-learning architecture for the given values and verify the performance of the system for the considered input data.

System requirement:

* Personal computer (PC) with sufficient RAM
* Preinstalled with necessary software (PYTHON)

**Theory:**

Neural network architecture refers to the arrangement and configuration of the various layers and components in a neural network model. Neural networks are a class of machine learning models inspired by the structure and function of the human brain. They consist of interconnected nodes (neurons) organized into layers, each layer serving a specific purpose in the data transformation process.

Deep neural networks (DNNs) are neural networks with multiple layers, often referred to as "deep" architectures. The depth of a neural network refers to the number of hidden layers it has between the input and output layers. DNNs have become increasingly popular due to their ability to learn intricate features and representations from complex data, allowing them to excel in a wide range of tasks such as image recognition, natural language processing, and more.

The number of neurons and layers in a deep neural network can significantly influence its performance, training time, and generalization capability. However, determining the optimal number of neurons and layers is a complex task that depends on various factors.

In practice, it's common to start with a moderately sized architecture and gradually increase its complexity until you achieve satisfactory results on validation data. Random search can be used to explore different hyperparameter settings, including the number of layers and neurons.

**Procedure:**

Step 1: Fix the NN architecture to be implemented to classy the data.

Step 2: Open the online simulation workspace “neural network playground”

Step 3: Based on the requirement, select the impute data and the following values for other parameters, like number of hidden layers, hidden neurons, input data, learning rate and activation.

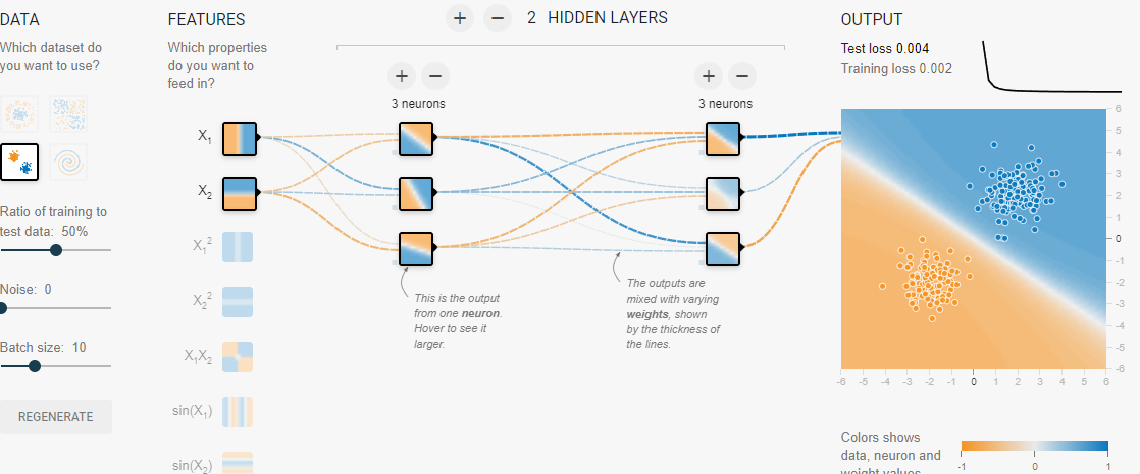
Step 4: Don’t modify any other parameters other than the parameters specified in step 3.

Step 5: Initiate the simulation using Play button and observe the classification result.

Step 6: Continuously check whether the expected classification result is achieved. If achieved, stope the execution and record the Epochs. If not achieved, stop the algorithms execution when the number of Epochs exceeds 200.

Experimental Result:

This section presents the experimental outcome of this work and the achieved result is presented below for a chosen database and a chosen DNN architecture.



Observation:

The experimental work is executed on the chosen database using the chosen NN scheme. This result confirms that the proposed scheme work well on the chosen image database and helps to achieve better detection accuracy.