

Assignment-3

Neural Network Playground

Vyas Milind Tarkeshkumar (2121721)

University Canada West

BUSI -651: Predictive Analytics: What Works?

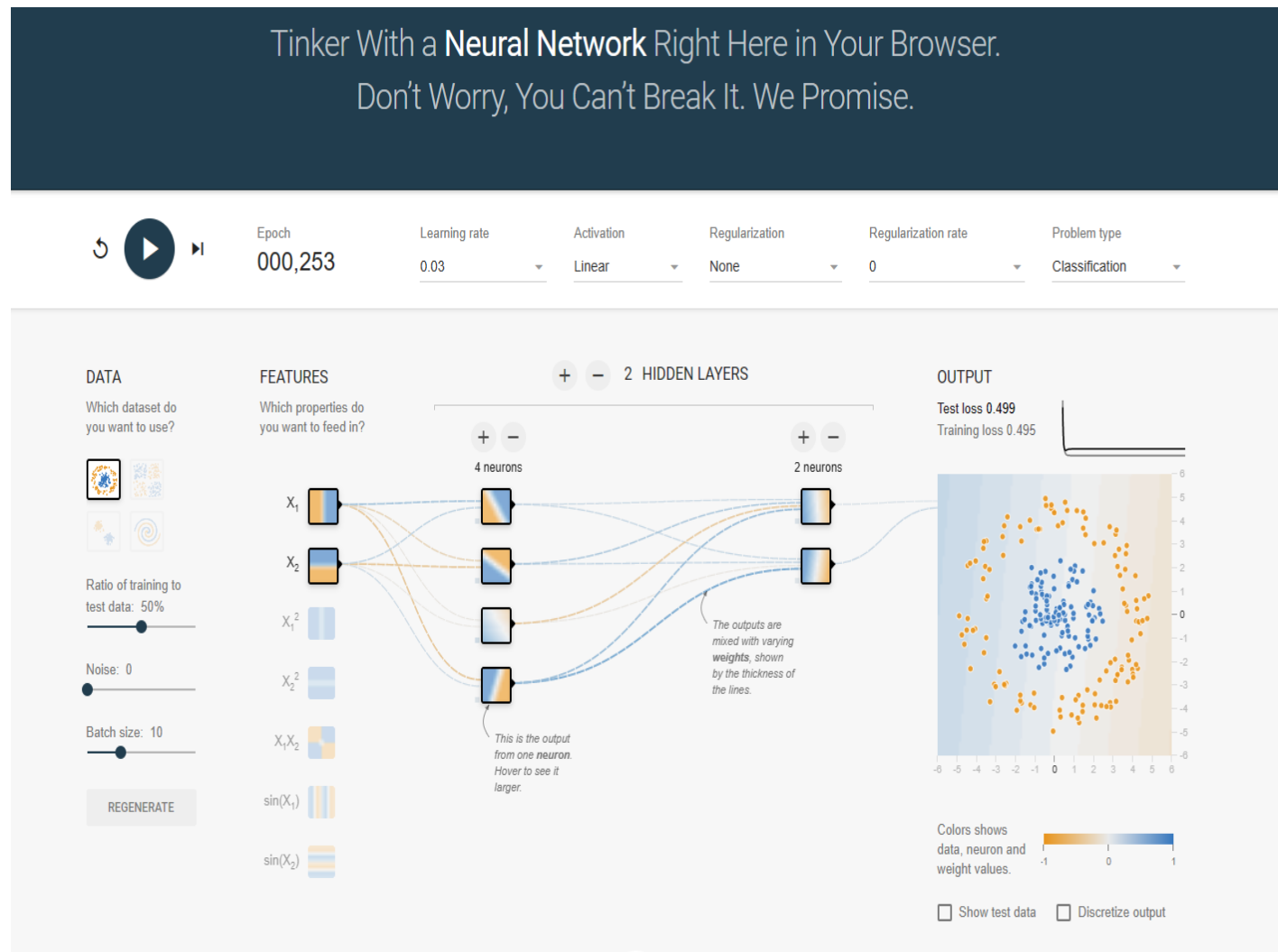
Mohsenijam Arash

March 17th, 2023

Introduction

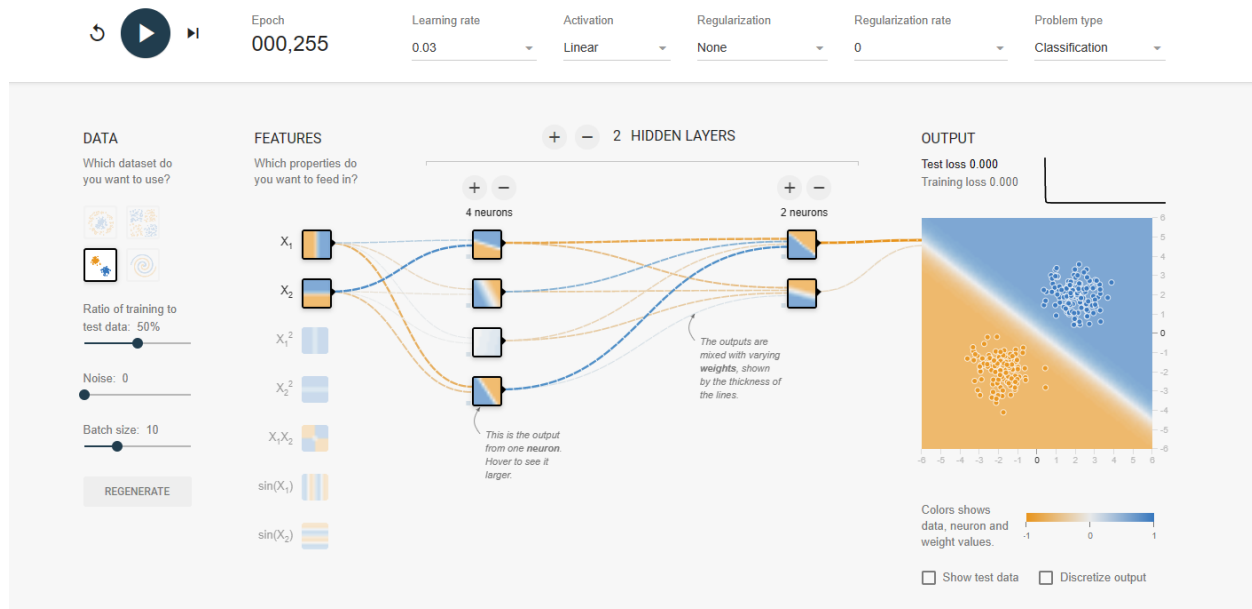
In this report, we will try to understand, how the neural networks work on feeding the datasets and analyzing effects by playing different models such as Circles, Gaussian, and Spirals to check whether the selected models fit into the datasets provided.

Task-1: Analyzing the effects of the activation function



Task-1: Creating a Neural Network playground using Circular data distribution and linear activation function.

Tinker With a **Neural Network** Right Here in Your Browser.
Don't Worry, You Can't Break It. We Promise.

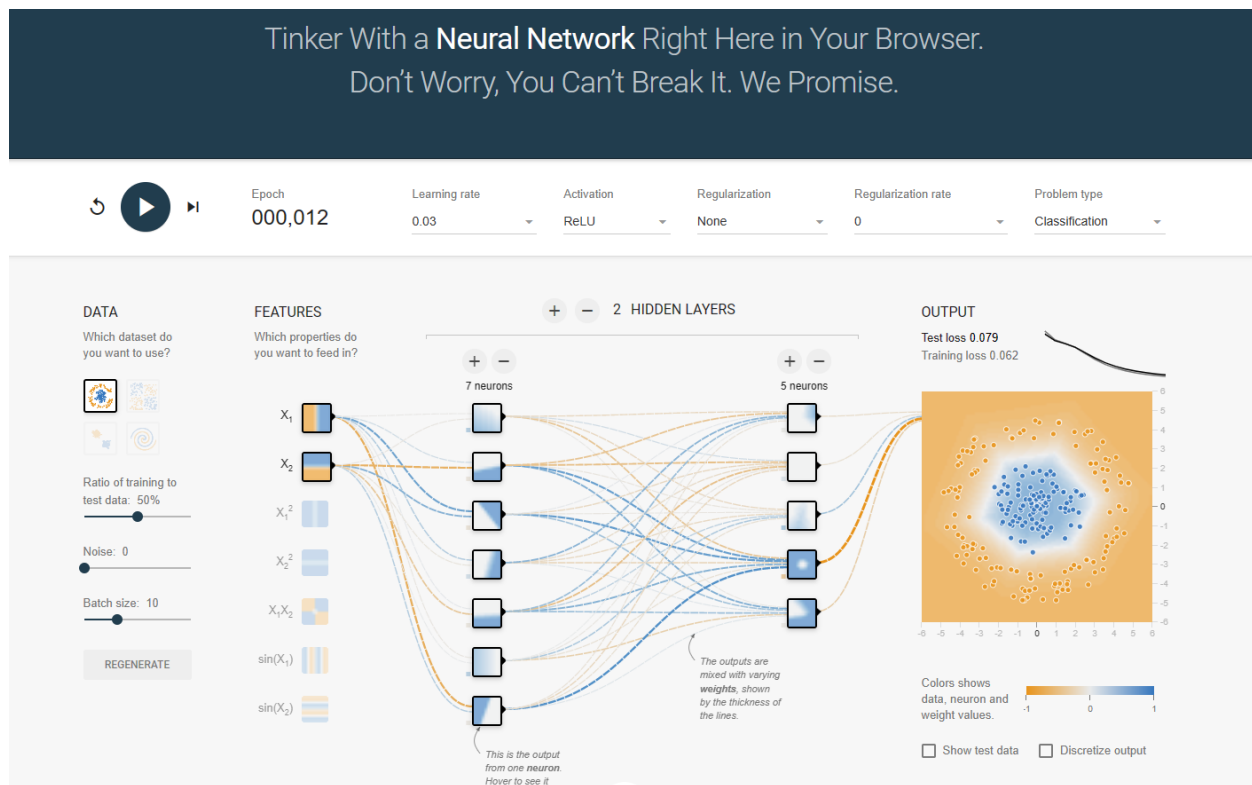


Task-1: Creating a Neural Network playground using **Gaussian data distribution and linear activation function**

Observation: Task-1

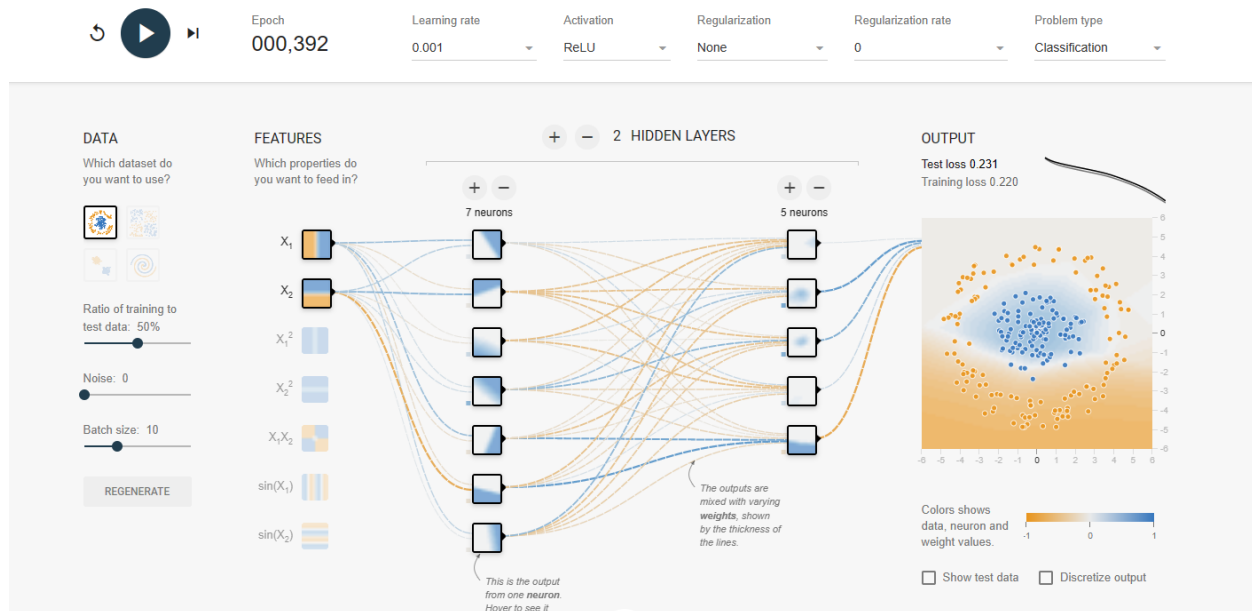
After training Circular and Gaussian models to 253 and 255 Epochs, a considerable convergence was seen in the case of the circular model but that was not the case with the Gaussian model. After applying filters as instructed, the **Test loss for the Circular model was 0.499** and for the **Gaussian model was 0.000**. This simply means that a combination of 2-hidden layers and 6-artificial neurons were enough for classifying the dataset using the Circular model but in the case of the Gaussian, model neurons were not enough to give data classification using this model.

Task 2: Effect of Learning Rate and Batch size



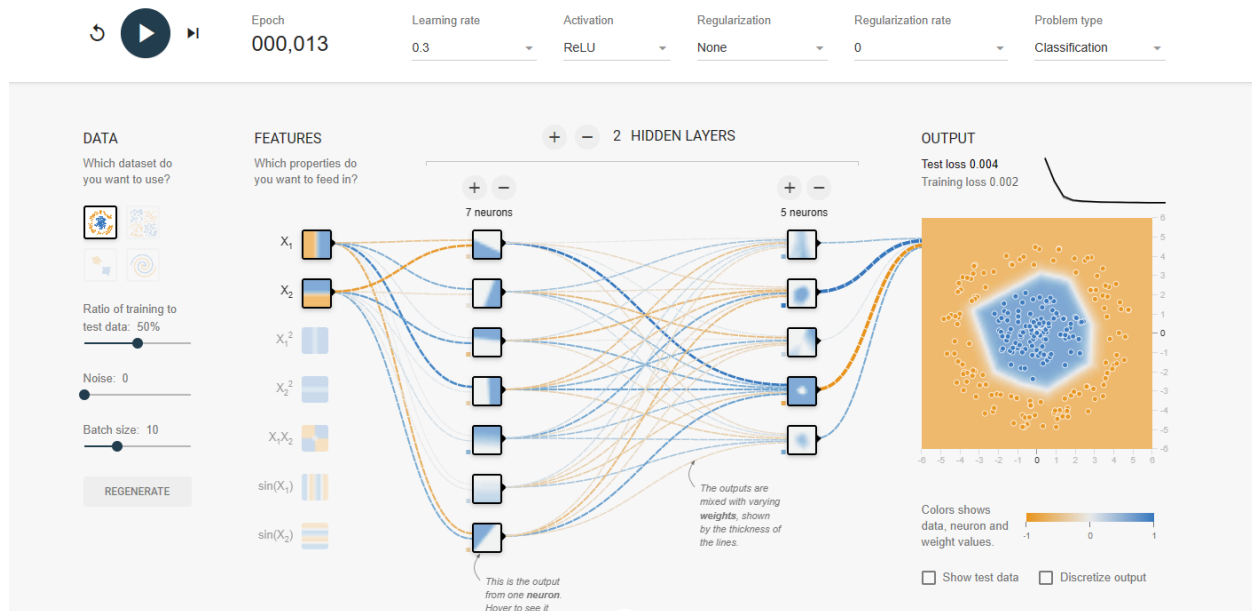
Task-2: Creating a Neural Network playground using Circular data distribution and ReLU activation function, Learning Rate: 0.03.

Tinker With a **Neural Network** Right Here in Your Browser.
Don't Worry, You Can't Break It. We Promise.



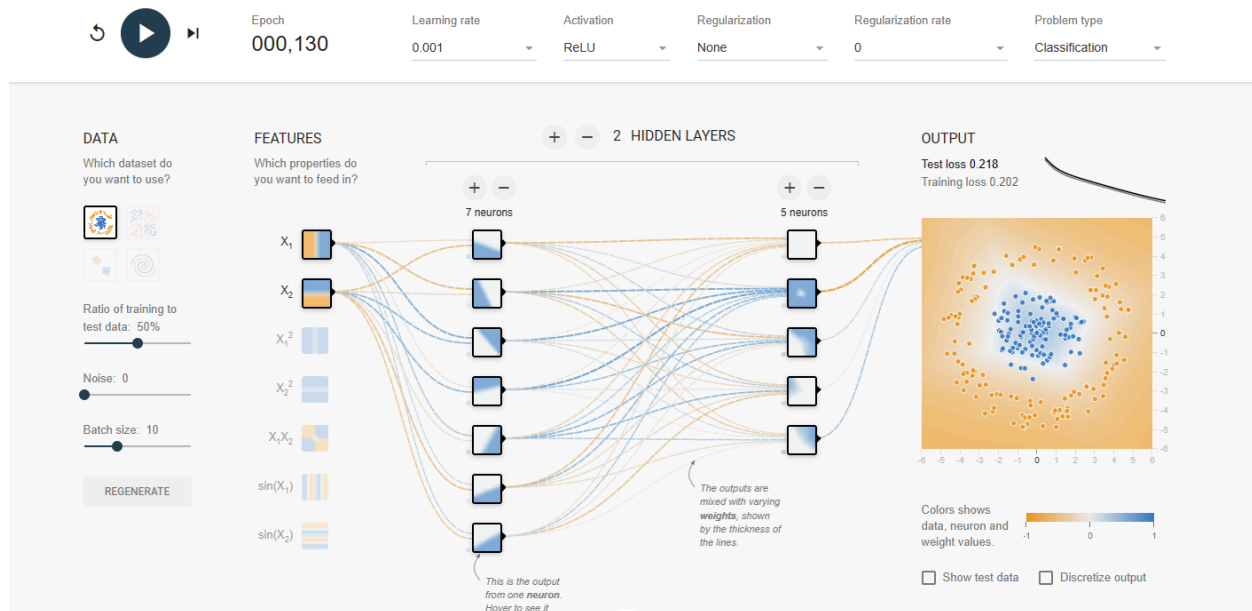
Task-2: Creating a Neural Network playground using **Circular data distribution** and **ReLU** activation function, **Learning Rate: 0.001**.

Tinker With a **Neural Network** Right Here in Your Browser.
Don't Worry, You Can't Break It. We Promise.



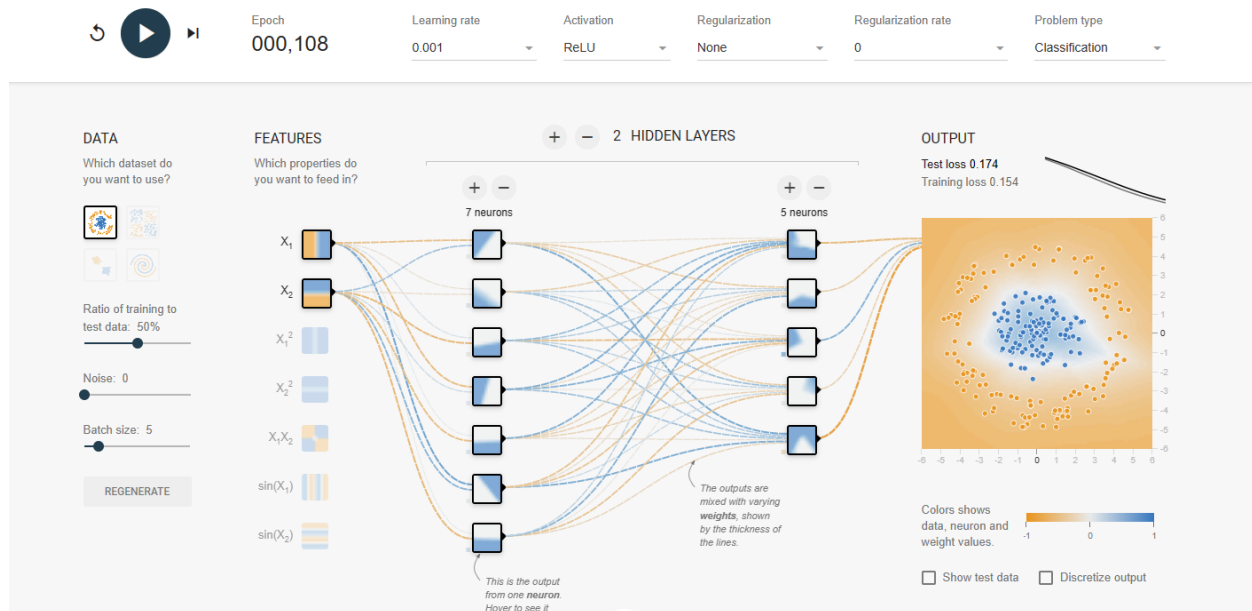
Task-2: Creating a Neural Network playground using **Circular data distribution** and **ReLU** activation function, **Learning Rate: 0.3**.

Tinker With a **Neural Network** Right Here in Your Browser.
Don't Worry, You Can't Break It. We Promise.



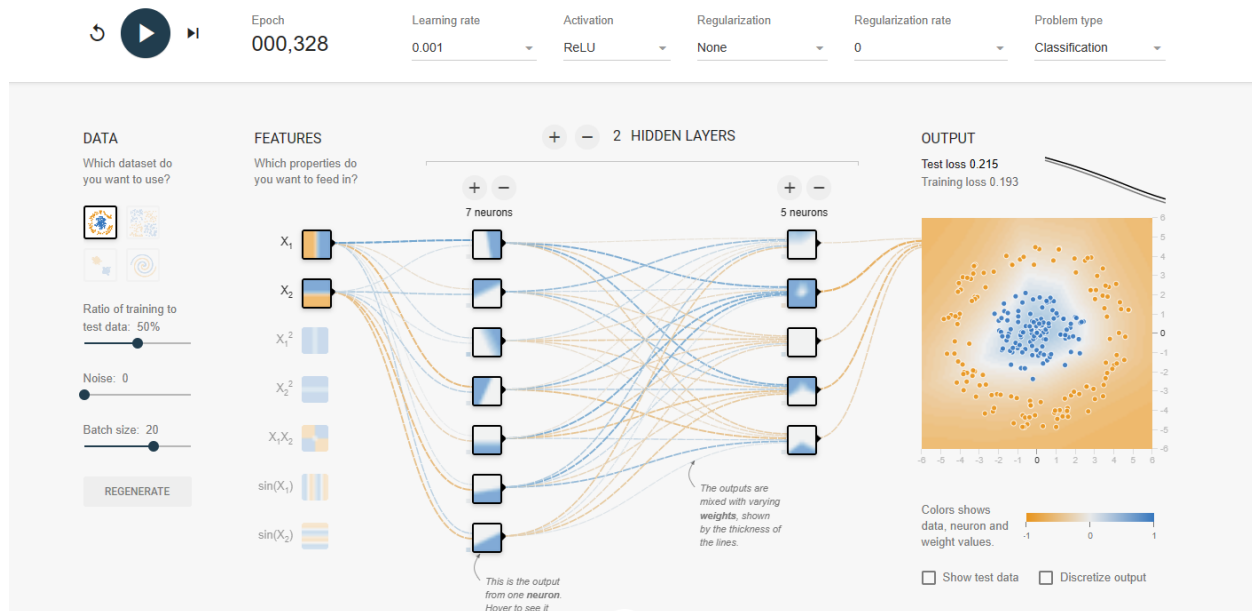
Task-2: Creating a Neural Network playground using **Circular data distribution** and **ReLU** activation function, **Learning Rate: 0.001**.

Tinker With a **Neural Network** Right Here in Your Browser.
Don't Worry, You Can't Break It. We Promise.



Task-2: Creating a Neural Network playground using **Circular data distribution and ReLU** activation function, **Learning Rate: 0.001, Batch Size: 05**.

Tinker With a **Neural Network** Right Here in Your Browser.
Don't Worry, You Can't Break It. We Promise.

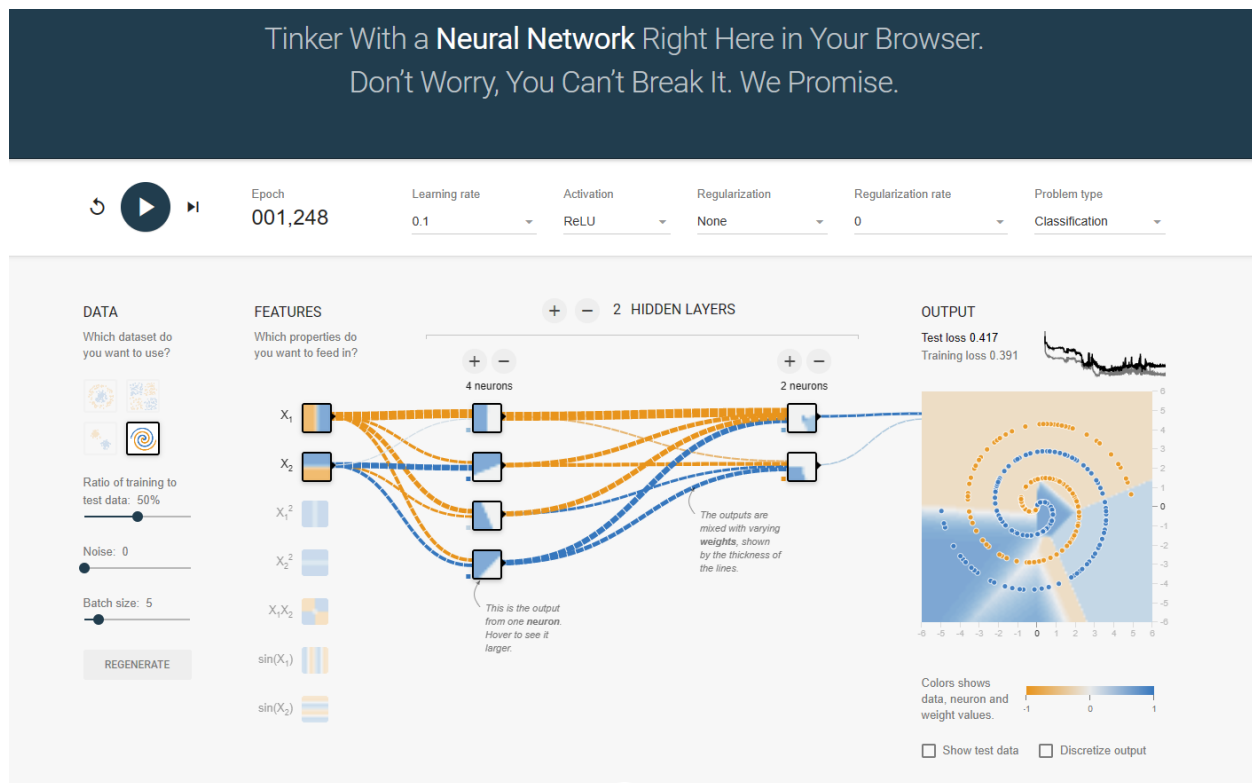


Task-2: Creating a Neural Network playground using **Circular data distribution and ReLU activation function, Learning Rate: 0.001, Batch Size: 20.**

Observation: Task-2

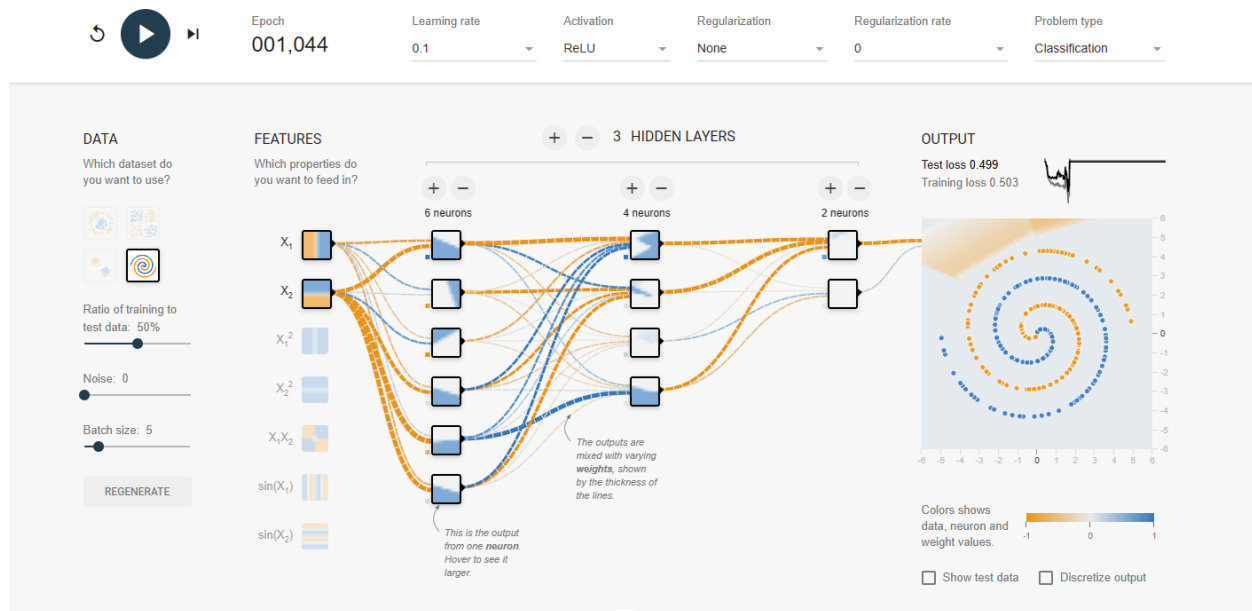
At a learning rate of 0.03 number of epochs was 012 but by making it to 0.3 number of epochs taken to learn was 013. Again dropping the learning rate to 0.001 number of epochs to learn increased to 108. It simply higher learning rate gives faster convergence and vice-versa. But by increasing batch size it takes more epochs for a model to learn data. For a learning rate of 0.001 test loss was 0.2 but for a rate of 0.3, it was 0.07. This implies decreasing the learning rate results in higher test loss for circular data distribution.

Task 3: Role of a more complex model



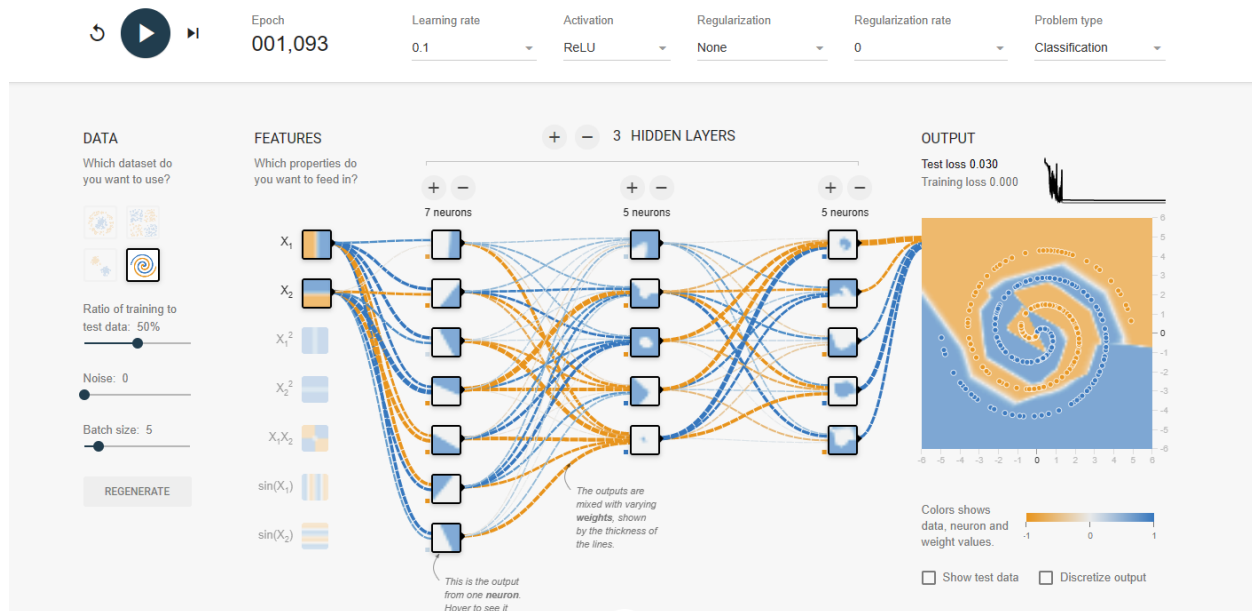
Task-3: Creating a Neural Network playground using **Spiral data distribution** and **ReLU** activation function, **Learning Rate: 0.1**, **Batch Size: 05**.

Tinker With a **Neural Network** Right Here in Your Browser.
Don't Worry, You Can't Break It. We Promise.



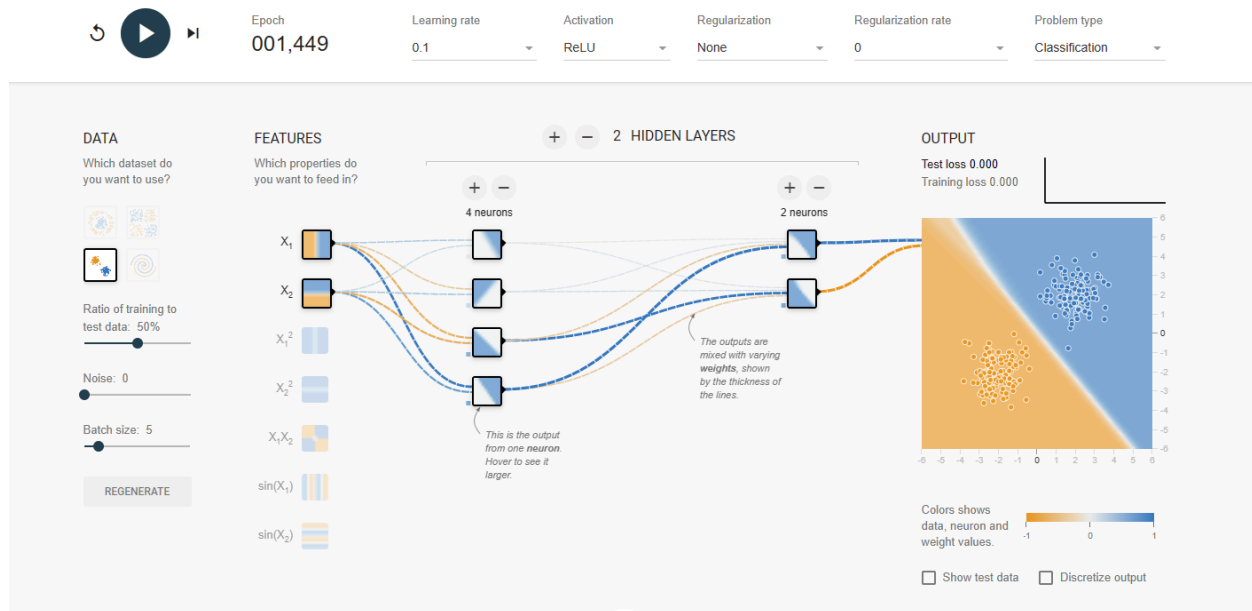
Task-3: Creating a Neural Network playground using **Spiral data distribution** and **ReLU** activation function, **Learning Rate: 0.1**, **Batch Size: 05**.

Tinker With a **Neural Network** Right Here in Your Browser.
Don't Worry, You Can't Break It. We Promise.



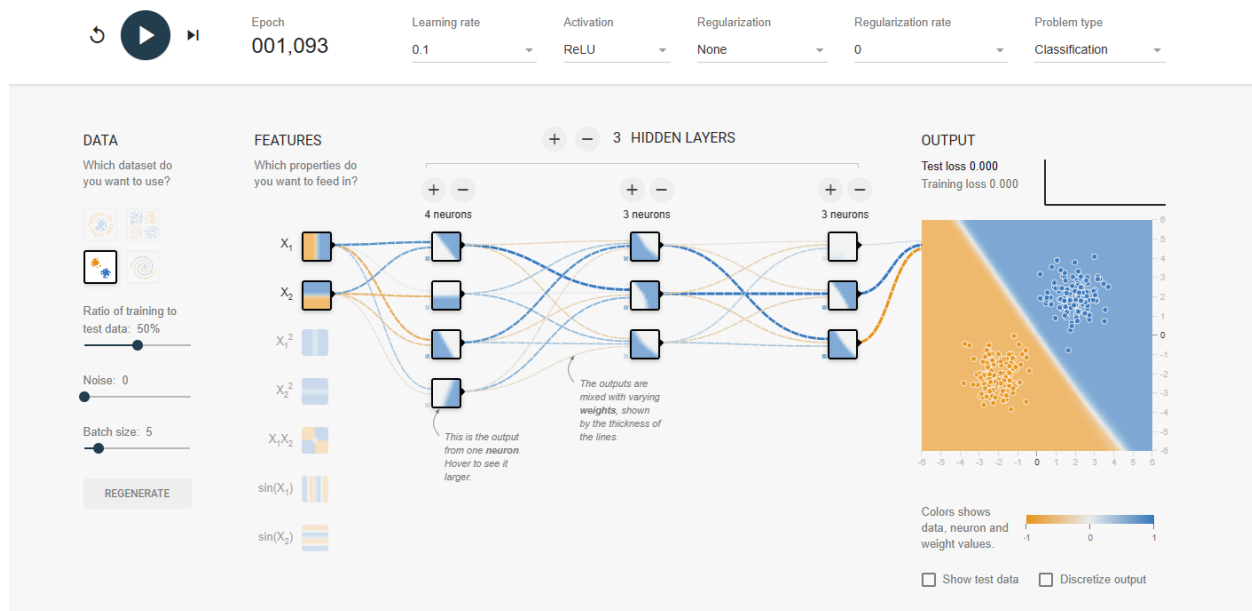
Task-3: Creating a Neural Network playground using **Spiral data distribution** and **ReLU** activation function, **Learning Rate: 0.1**, **Batch Size: 05**.

Tinker With a **Neural Network** Right Here in Your Browser.
Don't Worry, You Can't Break It. We Promise.



Task-3: Creating a Neural Network playground using **Gaussian data distribution** and **ReLU** activation function, **Learning Rate: 0.1**, **Batch Size: 05**.

Tinker With a **Neural Network** Right Here in Your Browser.
Don't Worry, You Can't Break It. We Promise.

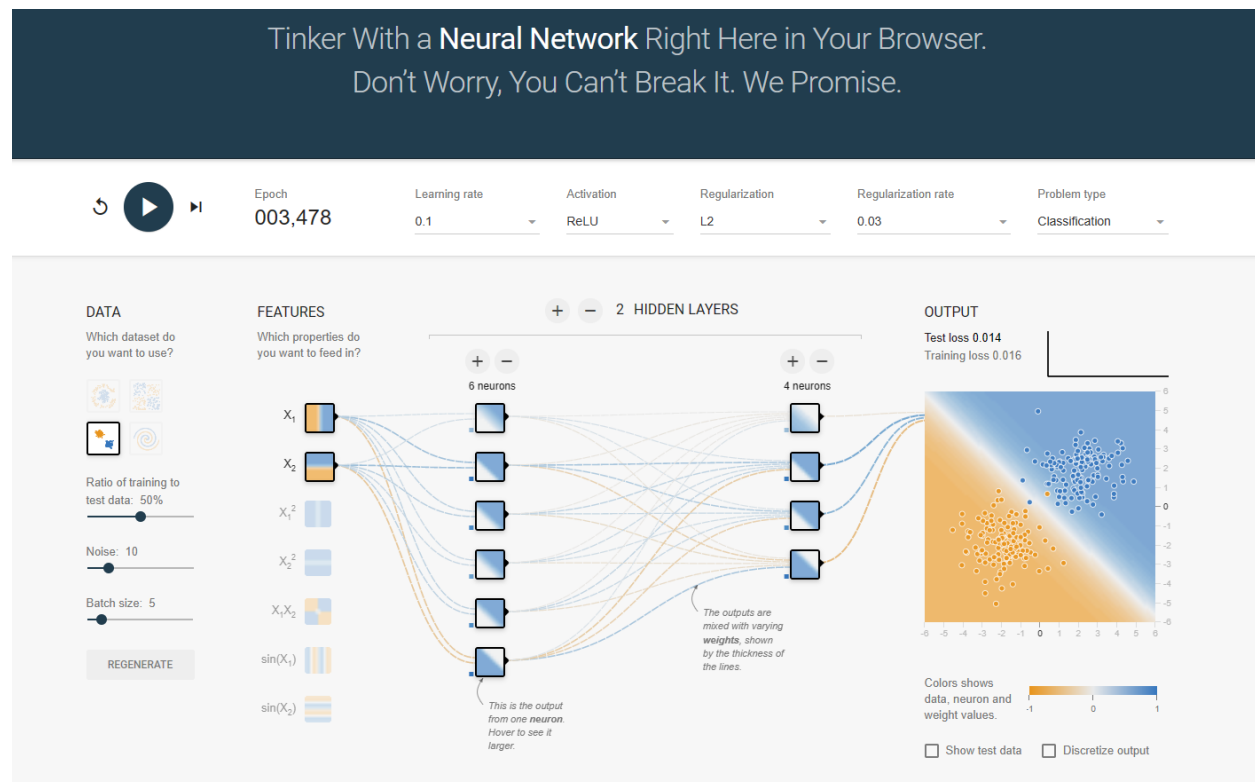


Task-3: Creating a Neural Network playground using **Gaussian data distribution and ReLU activation function, Learning Rate: 0.1, Batch Size: 05.**

Observation: Task-3

After training the data model even at 1500 epochs spiral model was unable to converge due to not enough neurons and hidden layers. **After feeding the 17 artificial neurons and 03 hidden layers, the model was finally able to converge into a spiral shape.** However, in the case of the Gaussian model, it was unable to fit into the provided dataset even after 03 layers of hidden input and 10 neurons. **This all implies the number of neurons plays a major role in classifying datasets in form of a neural network model. A more complex model will give better results for data classification.**

Task 4: Effect of Noise and Regularization

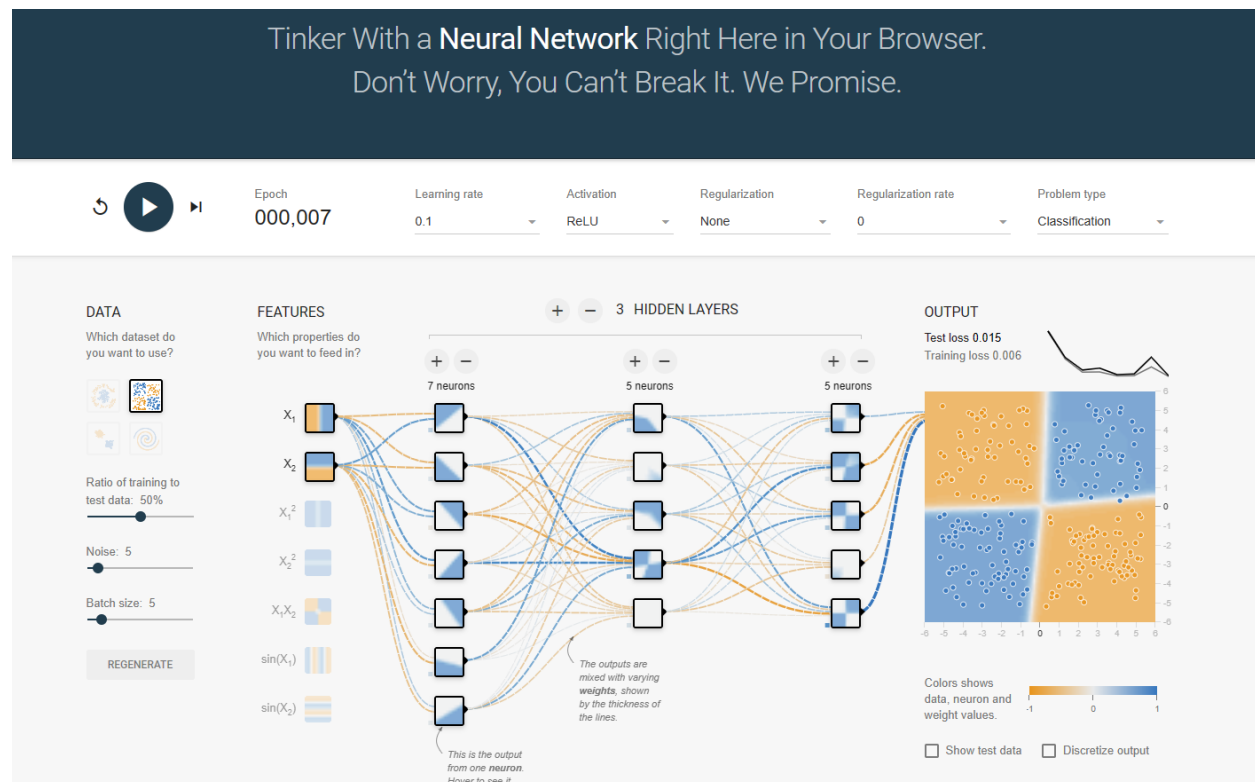


Task-4: Creating a Neural Network playground using **Gaussian data distribution** and **ReLU** activation function, **Learning Rate: 0.1**, **Batch Size: 05**, **Regularization level: L2**, **Regularization Rate: 0.03**, **Noise level: 10**.

Observation: Task-4

As discussed in the earlier part Gaussian, the model is not fit for this dataset. After training the data model to 3500 epoch with 2 hidden inputs layers and 10 neurons no change was observed in comparison to task-1 and task-3: step-8. In choosing a model, I will choose a model with the noise level and regularization rates as it gives me little opportunity to fit my model into a given dataset.

Task-5



Task-5: Creating a Neural Network playground using **Exclusive data distribution** and **ReLU** activation function, **Learning Rate: 0.1**, **Batch Size: 05**, **Regularization level: L2**, **Regularization Rate: 0.03**, **Noise level: 10**.

Observation and learning: Task-5

Here, I used exclusive data distribution and after combining all the filters, I found that with a learning rate of 0.1, it took only 007 epochs for a model to learn the dataset due to model complexity. One thing to take note of here is there is a minimal test loss of 0.015 in the process of Learning.

Conclusion

To conclude, I can say that a more complex model is a higher chance of learning with lower test loss for any of the data distribution models.