Deep Learning for Business

Deep Learning Project with TensorFlow Playground

Project Setup

Project Setup

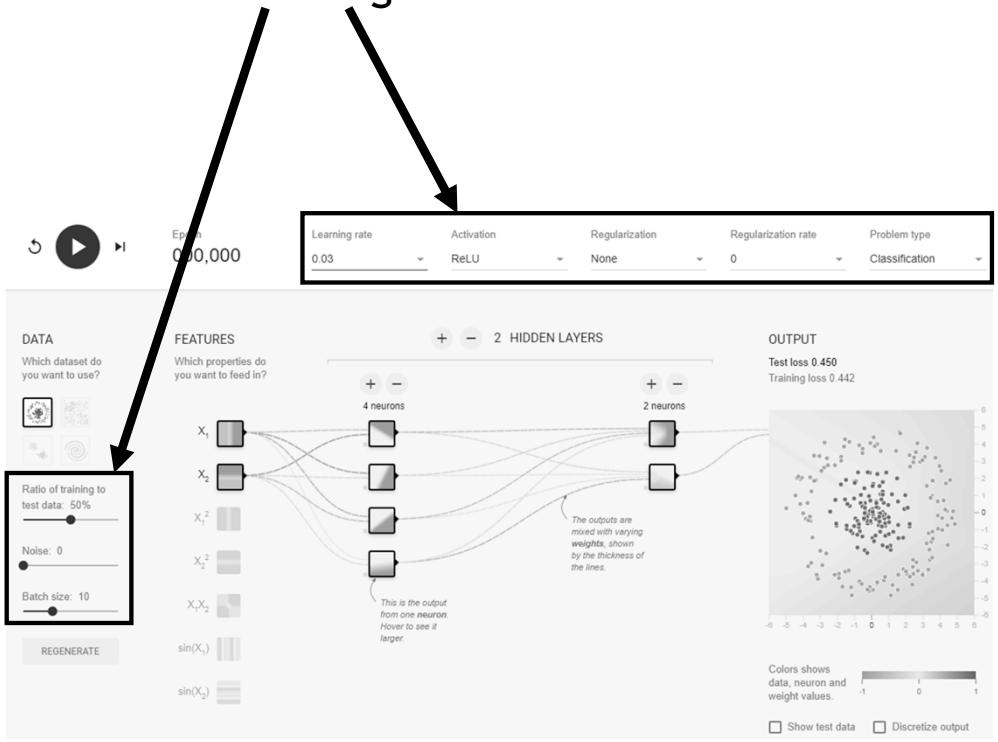
- TensorFlow Playground Project Setup
  - Learning rate : 0.03
  - Activation : ReLU
  - Regularization : None
  - Regularization rate : 0
  - Problem type : Classification
  - Ratio of training to test data : 50%
  - Noise : 0
  - Batch size : 10

Regularization is not needed in solving a simple problem, because overfitting most likely will not occur

Noise is set to zero to make it easy to find the solution. Try to practice more with higher noise levels

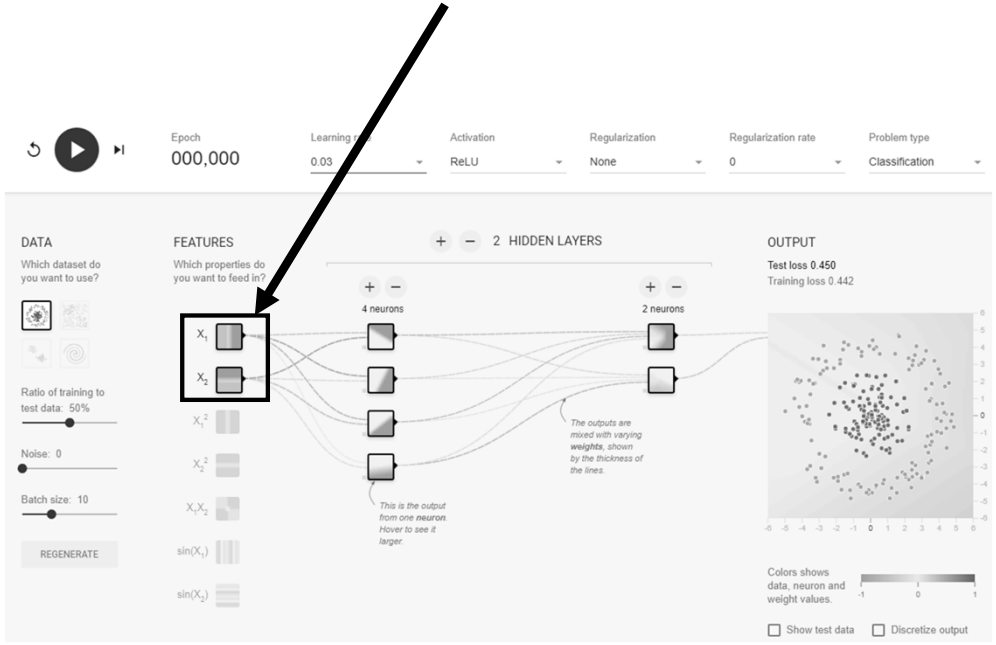
Project Environment

— The screen setting will then look like this



Project Environment

- The screen setting will then look like below
- We will use the X_1 and X_2 features



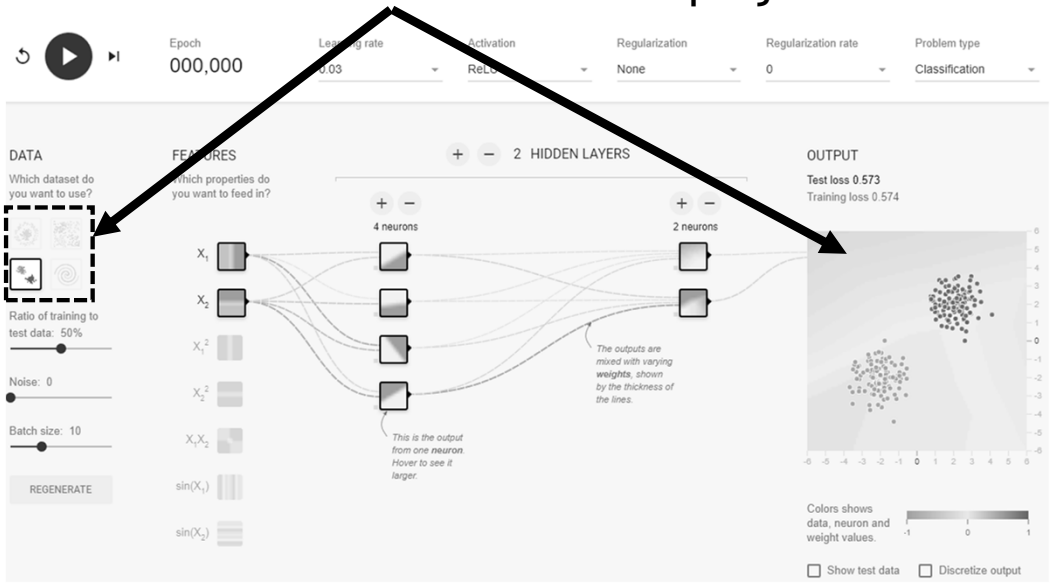
Deep Learning for Business

Deep Learning Project with TensorFlow Playground

Project 1

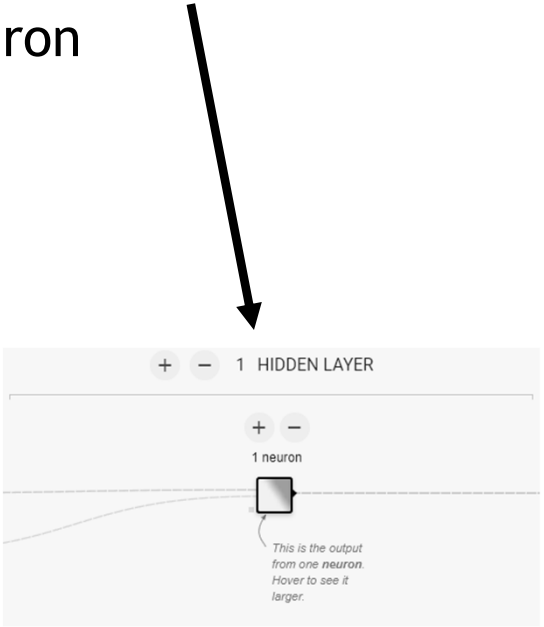
Project 1

- Objective of Project 1
  - Classification (separate) two sets of clusters of data
  - Select the data set for this project



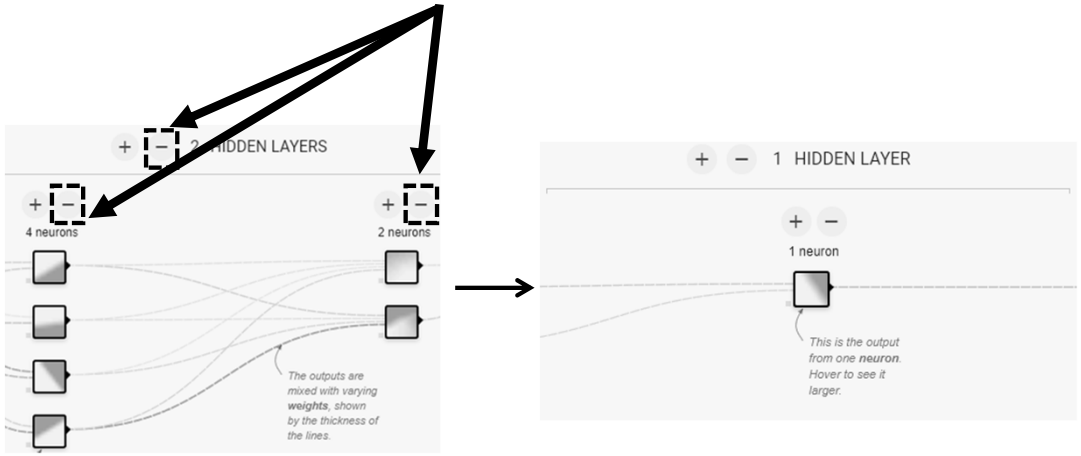
Project 1

- We will start with only one Hidden Layer that has only one Neuron



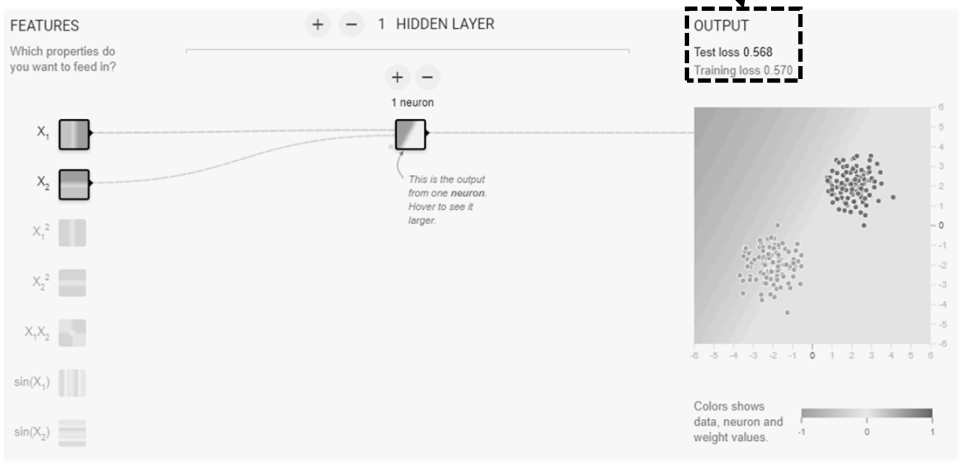
Project 1

- We will start with only one Hidden Layer that has only one Neuron
  - If you have more Hidden Layers and Neurons, then use the minus buttons to remove them



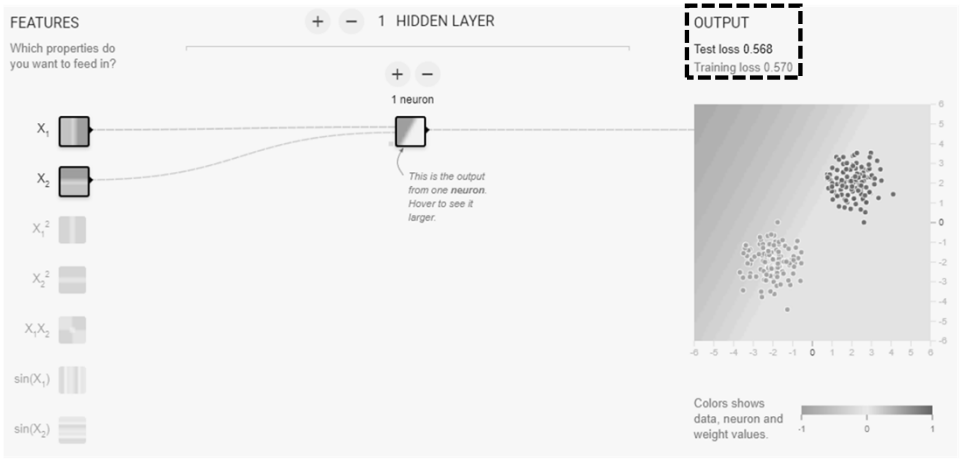
Project 1

- Because the NN training has not started, the Test Loss and Training Loss are very high



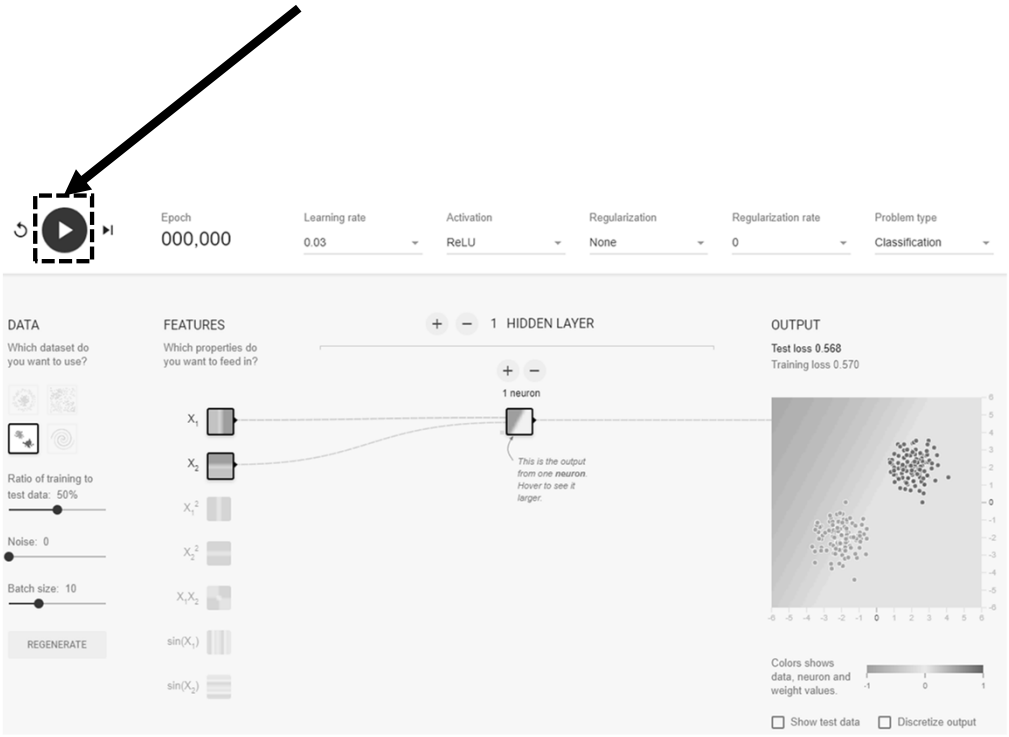
Project 1

- Because the NN training has not started, the Test Loss and Training Loss are very high
- Initial Test Loss and Training Loss values will be different because the initial weight value is set at random



Project 1

– Press the run ▶ button to start the training

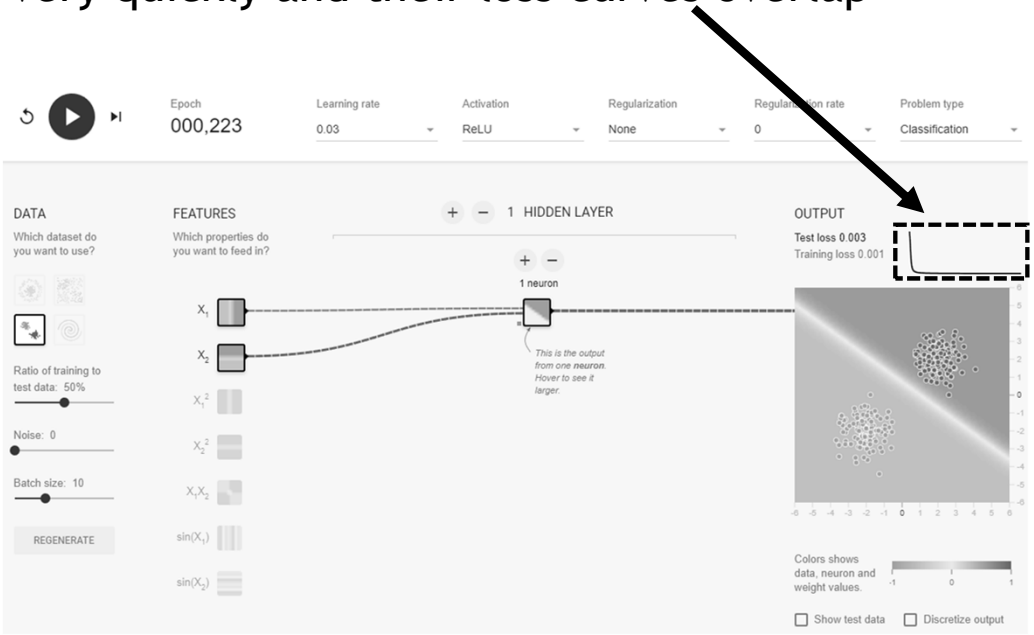


Project 1

- Since this is a simple problem, it will end fast successfully

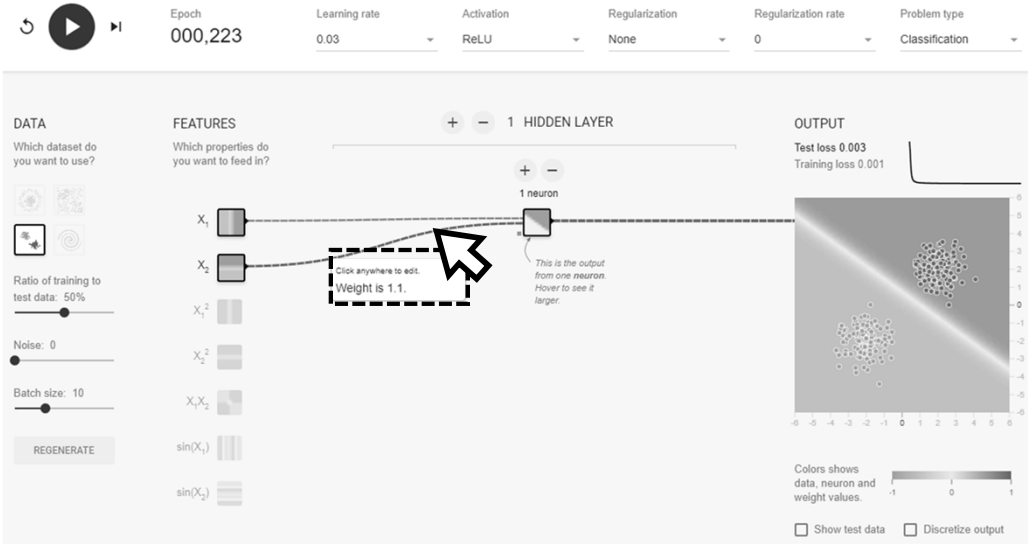
Project 1

- Since this is a simple problem, it will end fast successfully
- Test Loss & Training Loss values become very small very quickly and their loss curves overlap



Project 1

- Check the **weight** value by placing the **mouse** cursor on the lines between the layers

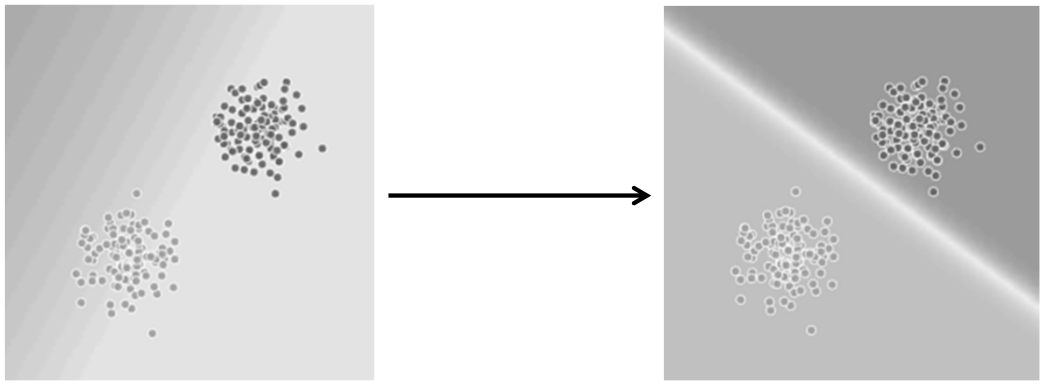


Project 1

- Before training, the NN could not distinguish between the **orange** and **blue** data points

Project 1

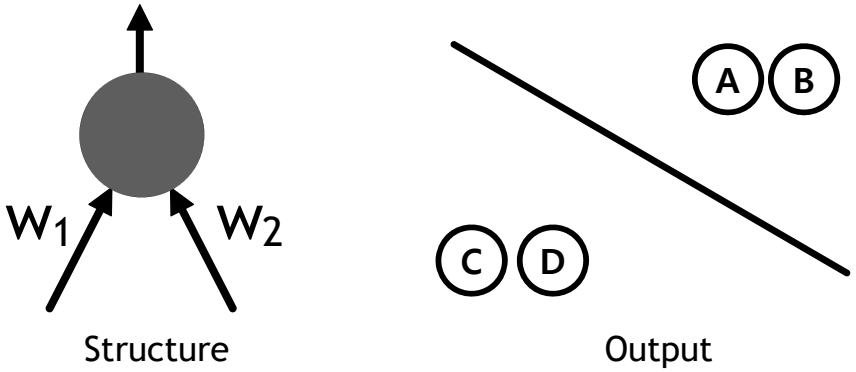
- Before training, the NN could not distinguish between the **orange** and **blue** data points
- After training, the **orange** and **blue** regions are perfectly distinguished



Project 1

- This problem can be solved with one hidden layer neuron
- We learned about this in Module 4

  - Single layer with weights: w_1 w_2



Martin T. Hagan, Howard B. Demuth, Mark H. Beale, Orlando De Jesús, Neural Network Design, 2nd Ed. ISBN-13: 978-0971732117, Martin Hagan, Sept. 1, 2014.

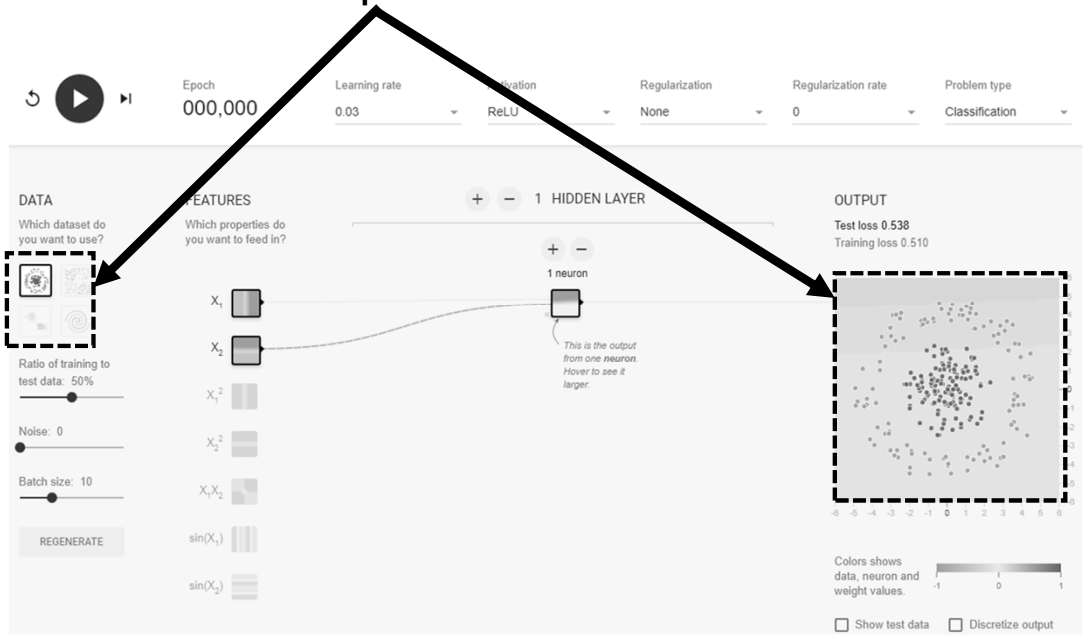
Deep Learning for Business

Deep Learning Project with TensorFlow Playground

Project 2

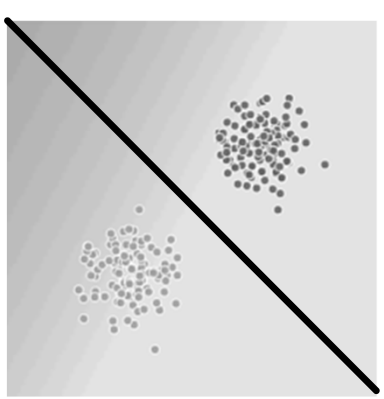
Project 2

- In project 2, we will distinguish the two data sets
  - Orange data surrounds the blue data in a circular shape

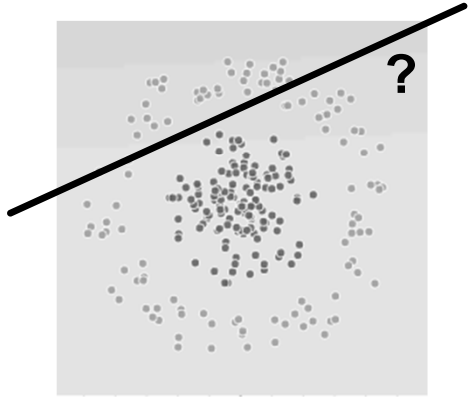


Project 2

- Project 2's problem is much more complicated than the problem in Project 1
- One line is not sufficient as a solution
  - Multiple neurons in the hidden layer are needed



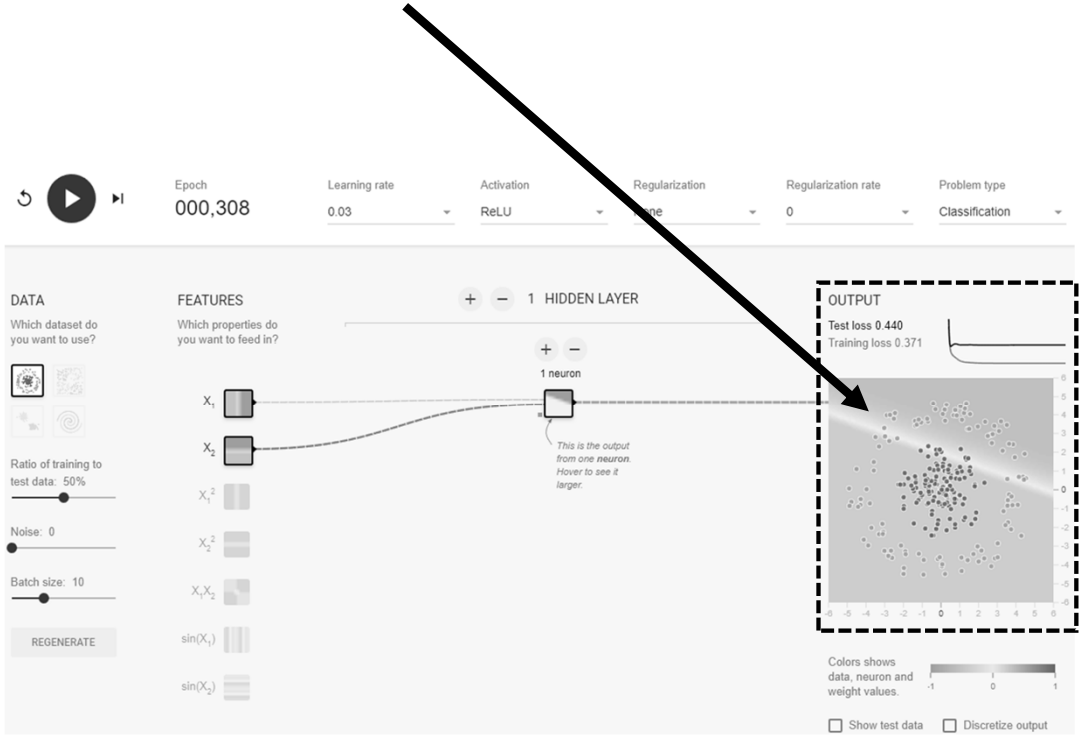
Project 1



Project 2

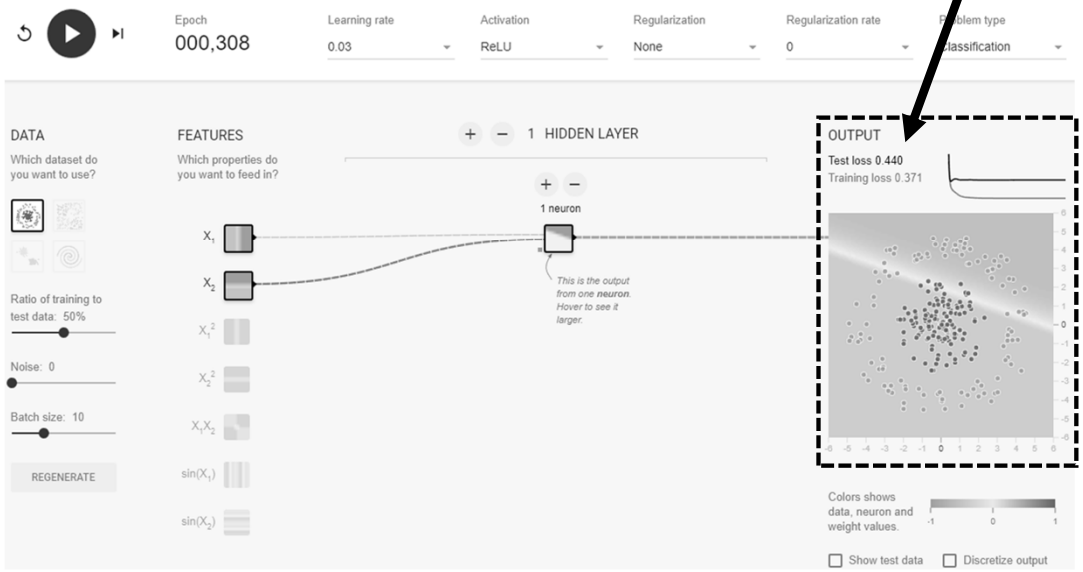
Project 2

— If you train with one hidden layer neuron, the classification fails



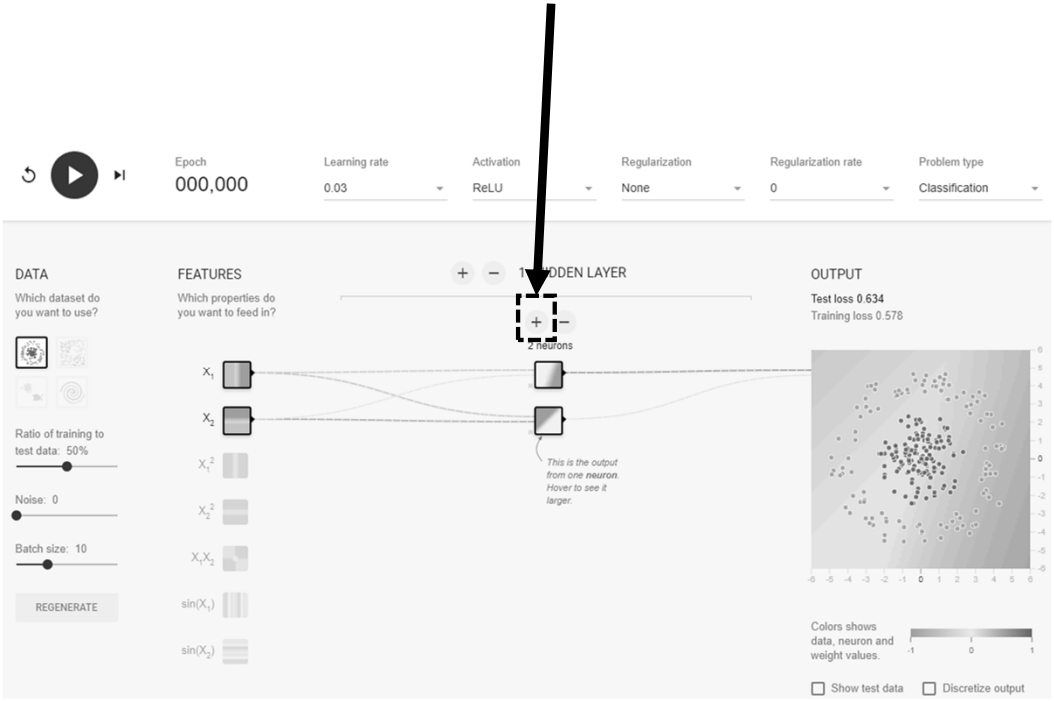
Project 2

- If you train with one hidden neuron, the classification fails
- Output results of Test Loss 0.440 and Training Loss 0.371 show that the classification has failed



Project 2

— Press the plus button to increase the number of hidden layer neurons to two

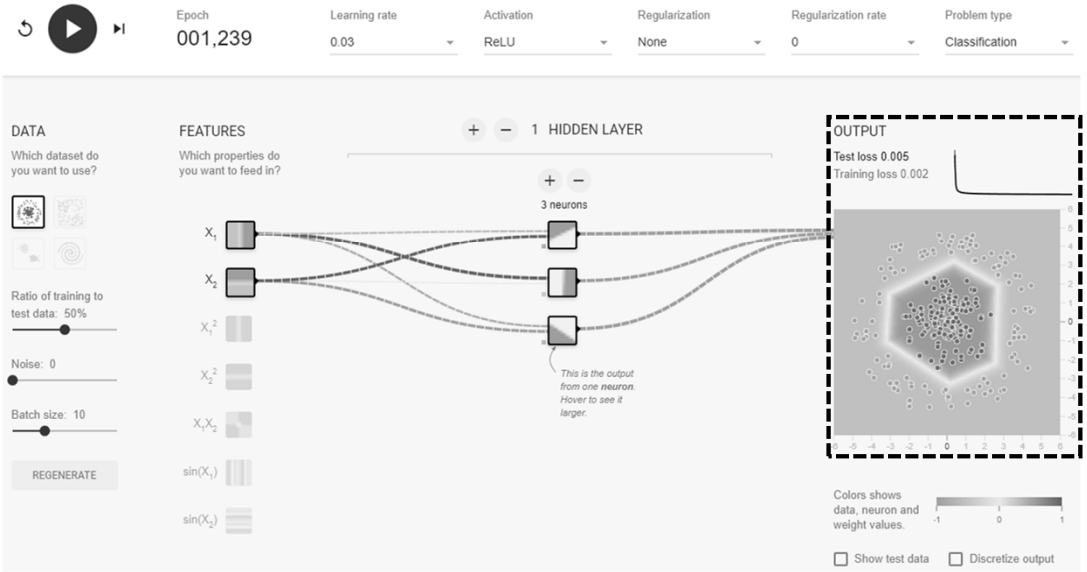


Project 2

- With two neurons, the performance improves
  - Test Loss 0.440 ➔ 0.270
  - Training Loss 0.371 ➔ 0.249
- But still the classification fails

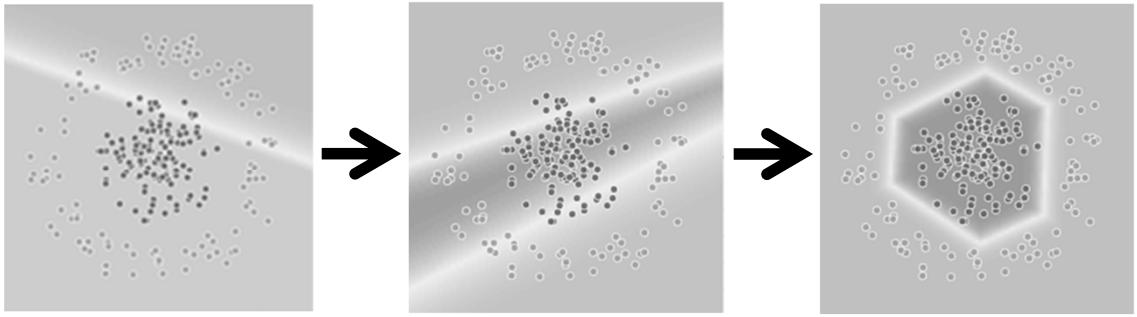
Project 2

- With three neurons, the performance improves
  - Test loss 0.440 ➔ 0.270 ➔ 0.005
  - Training loss 0.371 ➔ 0.249 ➔ 0.002
- Classification succeeds



Project 2

- Observe the difference in Classification performance when using 1, 2, and 3 neurons in the Hidden Layer of the NN



Deep Learning for Business

Deep Learning Project with TensorFlow Playground

References

References

- A Neural Network Playground – TensorFlow [Online]. Available: http://playground.tensorflow.org
- Google Cloud Big Data and Machine Learning Blog [Online]. Available: https://cloud.google.com/blog/big-data/2016/07/understanding-neural-networks-with-tensorflow-playground
- Google Open Source [Online]. Available: https://opensource.google.com/projects/tensorflow-playground