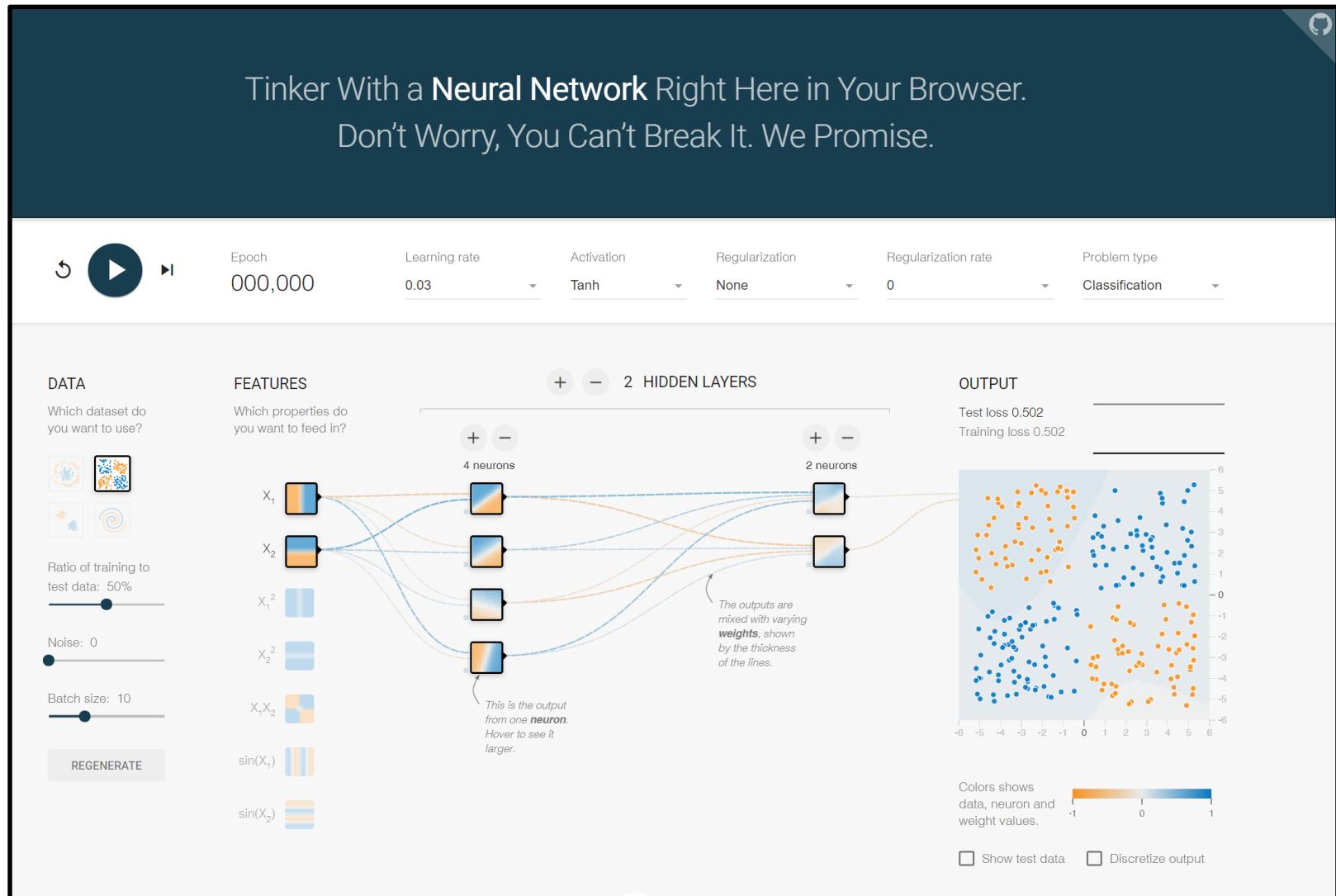


8 Machine Learning I

- Neural Networks & Deep Learning (3) TensorFlow Playground



8 Machine Learning I

- Neural Networks & Deep Learning

Content:

1. Motivation
2. Basics of
Neural Networks (NN)
3. TensorFlow Playground
4. Basics of Convolutional
Neural Networks (CNN)
5. Deep Learning (DL) by
Deepmind: AlphaGo, Zero...
6. Deep NN in
Tesla Autonomous Driving
7. Summary

0
1
2
3
4
5
6
7
8
9

8 Machine Learning I

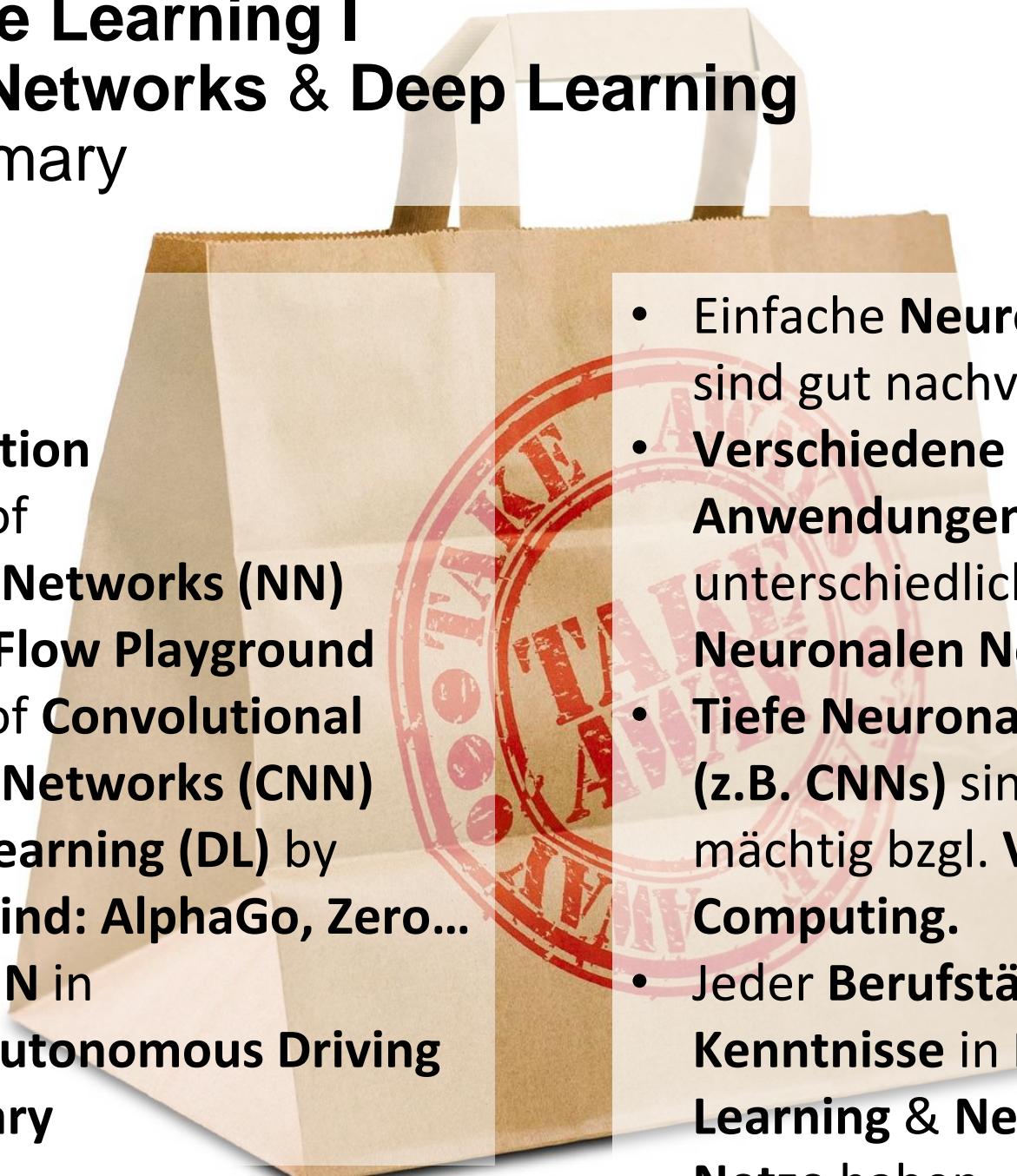
- Neural Networks & Deep Learning

(7) Summary

Content:

1. Motivation
2. Basics of Neural Networks (NN)
3. TensorFlow Playground
4. Basics of Convolutional Neural Networks (CNN)
5. Deep Learning (DL) by Deepmind: AlphaGo, Zero...
6. Deep NN in Tesla Autonomous Driving
7. Summary

- Einfache **Neuronale Netze** sind gut nachvollziehbar.
- Verschiedene Anwendungen benötigen unterschiedliche **Neuronale Netze**.
- Tiefe **Neuronale Netze** (z.B. CNNs) sind sehr mächtig bzgl. **Visual Computing**.
- Jeder **Berufstätige** sollte Kenntnisse in **Machine Learning & Neuronale Netze** haben.



Michael Amberg

Todays Content:

- 1. Motivation**
- 2. Basics of Neural Networks (NN)**
- 3. TensorFlow Playground**
- 4. Basics of Convolutional Neural Networks (CNN)**
- 5. Deep Learning (DL) by Deepmind: AlphaGo, Zero...**
- 6. Deep NN in Tesla Autonomous Driving**
- 7. Summary**



TensorFlow Playground

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

Epoch: 000,000 Learning rate: 0.03 Activation: Tanh Regularization: None Regularization rate: 0 Problem type: Classification

DATA
Which dataset do you want to use?

Ratio of training to test data: 50%
Noise: 0
Batch size: 10

FEATURES
Which properties do you want to feed in?

2 HIDDEN LAYERS
+ -
4 neurons 2 neurons
The outputs are mixed with varying weights, shown by the thickness of the lines.
This is the output from one neuron. Hover to see it larger.

OUTPUT
Test loss 0.502
Training loss 0.502

Colors shows data, neuron and weight values.
 Show test data Discretize output

TensorFlow Playground

Epoch 000,000 Learning rate 0.03 Activation Tanh Regularization None Regularization rate 0 Problem type Classification

DATA FEATURES OUTPUT

Which dataset do you want to use? Which properties do you want to feed in?

Test loss 0.502
Training loss 0.502

Ratio of training to test data: 50% Noise: 0 Batch size: 10

REGENERATE sin(X_1) sin(X_2)

2 HIDDEN LAYERS

+ -

4 neurons

2 neurons

X_1

X_2

X_1^2

X_2^2

$X_1 X_2$

This is the output from one neuron. Hover to see it larger.

The outputs are mixed with varying weights, shown by the thickness of the lines.

REGENERATE sin(X_1) sin(X_2)

Colors shows data, neuron and weight values. -1 0 1

Show test data Discretize output

The TensorFlow Playground is a web-based tool for visualizing and understanding neural network architectures. The interface includes a header with various configuration options like epoch, learning rate, activation function, and regularization. Below this, the 'DATA' section lists available datasets, and the 'FEATURES' section allows selecting input properties. A central feature is a neural network diagram with two hidden layers of 4 and 2 neurons respectively, receiving inputs X_1 , X_2 , X_1^2 , X_2^2 , and $X_1 X_2$. The diagram illustrates how multiple neurons in a layer produce different outputs, which are then weighted and summed. The 'OUTPUT' section displays a scatter plot of training data points (orange and blue) and the model's predictions (colored dots). A color bar at the bottom indicates the scale of data, neuron, and weight values. Various sliders and buttons for parameters like ratio of training/test data, noise, and batch size are also present.

TensorFlow Playground

Epoch 000,000 Learning rate 0.03 Activation Tanh Regularization None Regularization rate 0 Problem type Classification

DATA
Which dataset do you want to use?

Ratio of training to test data: 50%
Noise: 0
Batch size: 10
REGENERATE

FEATURES
Which properties do you want to feed in?
 X_1 X_2 X_1^2 X_2^2 $X_1 X_2$ $\sin(X_1)$ $\sin(X_2)$

2 HIDDEN LAYERS
+ - 4 neurons + - 2 neurons
The outputs are mixed with varying **weights**, shown by the thickness of the lines.
This is the output from one **neuron**. Hover to see it larger.

OUTPUT
Test loss 0.502 Training loss 0.502

Colors shows data, neuron and weight values.
 Show test data Discretize output

TensorFlow Playground

Epoch 000,000 Learning rate 0.03 Activation Tanh Regularization None Regularization rate 0 Problem type Classification

DATA
Which dataset do you want to use?


Ratio of training to test data: 50%
Noise: 0
Batch size: 10

FEATURES
Which properties do you want to feed in?
 X_1 X_2 X_1^2 X_2^2 $X_1 X_2$
 $\sin(X_1)$ $\sin(X_2)$

2 HIDDEN LAYERS
4 neurons
2 neurons

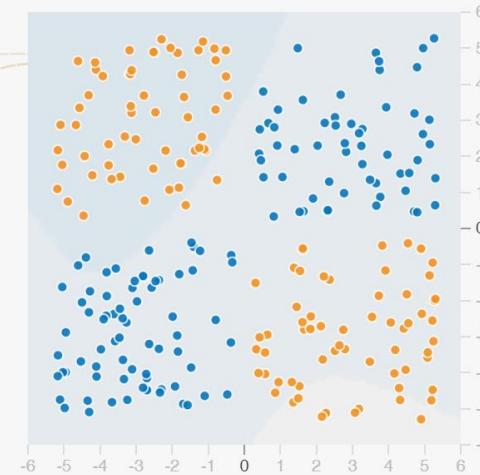
OUTPUT
Test loss 0.502
Training loss 0.502

The outputs are mixed with varying **weights**, shown by the thickness of the lines.

This is the output from one **neuron**. Hover to see it larger.

REGENERATE

Colors shows data, neuron and weight values.
 Show test data Discretize output



TensorFlow Playground

Epoch 000,000 Learning rate 0.03 Activation Tanh Regularization None Regularization rate 0 Problem type Classification

DATA Which dataset do you want to use?  

FEATURES Which properties do you want to feed in?       

HIDDEN LAYERS 2

OUTPUT Test loss 0.502 Training loss 0.502

Ratio of training to test data: 50%

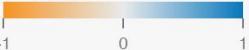
Noise: 0

Batch size: 10

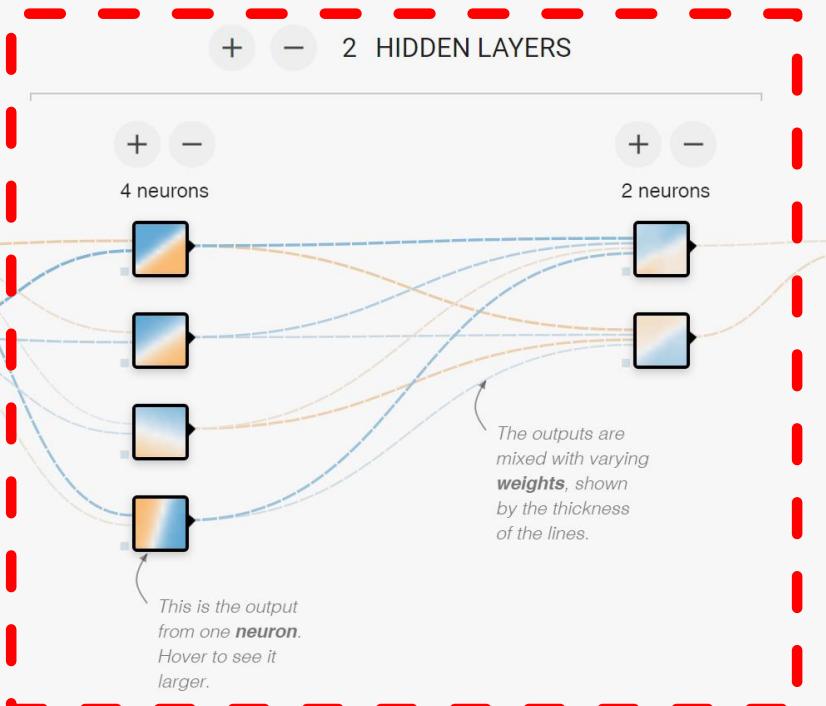
REGENERATE

The outputs are mixed with varying **weights**, shown by the thickness of the lines.

This is the output from one **neuron**. Hover to see it larger.

Colors shows data, neuron and weight values. 

Show test data Discretize output



TensorFlow Playground

Epoch 000,000 Learning rate 0.03 Activation Tanh Regularization None Regularization rate 0 Problem type Classification

DATA FEATURES + - 2 HIDDEN LAYERS

Which dataset do you want to use?
MNIST, Fashion MNIST, Iris, Spiral

Which properties do you want to feed in?
 X_1 , X_2 , X_1^2 , X_2^2 , $X_1 X_2$, $\sin(X_1)$, $\sin(X_2)$

Ratio of training to test data: 50%
Noise: 0
Batch size: 10

REGENERATE

OUTPUT
Test loss 0.502
Training loss 0.502

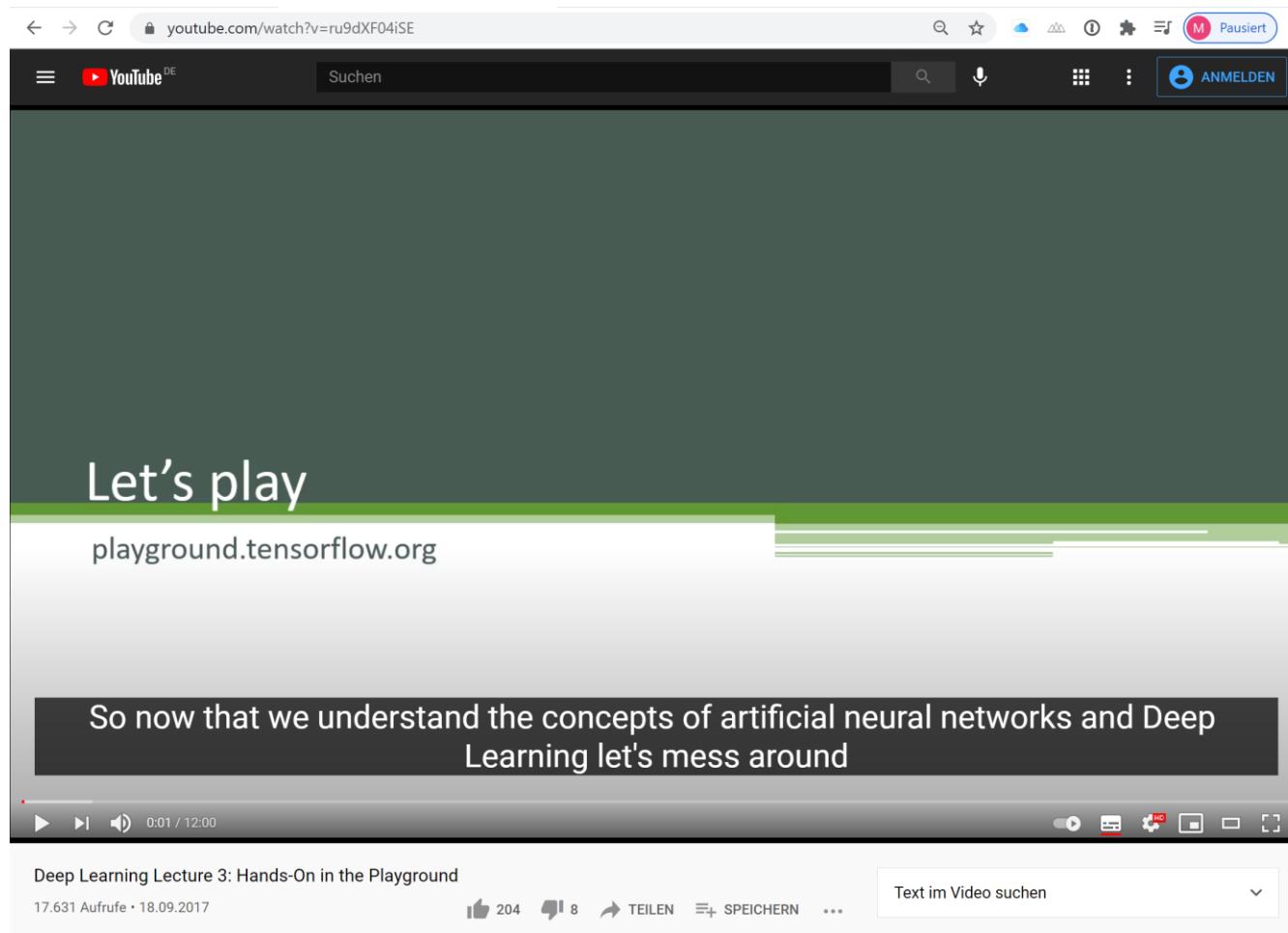
The outputs are mixed with varying **weights**, shown by the thickness of the lines.

This is the output from one **neuron**. Hover to see it larger.

Colors shows data, neuron and weight values.
Show test data Discretize output

The TensorFlow Playground is a web-based tool for visualizing and understanding neural network architectures. It allows users to experiment with different configurations and observe their effects on the model's performance and predictions. The interface includes sections for DATA selection, FEATURES definition, and a detailed view of the neural network structure with two hidden layers. A scatter plot on the right shows the distribution of training and test data points, with colors indicating data density and neuron/weight values. Various controls like learning rate, activation functions, and regularization are also present.

TensorFlow Playground Tutorial



Deep Learning Lecture 3: Hands-On in the Playground 2017 (12min)

<https://www.youtube.com/watch?v=ru9dXF04iSE>

TensorFlow Playground Help

The screenshot shows the TensorFlow Playground interface with the following settings:

- Epoch: 000,000
- Learning rate: 0.03
- Activation: Tanh
- Regularization: None
- Regularization rate: 0
- Problem type: Classification

The interface is divided into sections:

- DATA**: Shows a 2D scatter plot of blue and orange data points forming two spirals.
- FEATURES**: Displays input features X_1 and X_2 .
- HIDDEN LAYERS**: Shows two hidden layers, each with 2 neurons. A tooltip indicates: "The outputs are mixed with varying weights, shown by the colors of the lines." Another tooltip says: "This is the output from one neuron. Hover to see it larger."
- OUTPUT**: Shows test loss (0.502) and training loss (0.502).
- Bottom Controls**: Includes "REGENERATE", "sin(X_1)" and "sin(X_2)" buttons, a color bar for values from -1 to 1, and checkboxes for "Show test data" and "Discretize output".

Overlaid text on the interface includes:

- Lex Fridman: Neural networks learning spirals 2020**
www.youtube.com/watch?v=i3ZnDRrmFjg
- Understanding Deep Learning with TensorFlow playground 2018**
medium.com/@andrewt3000/understanding-tensorflow-playground-c20cdb7a250b
- Microsoft Word - Workshop Exercises with Tensorflow Playground.docx**
sigmaxi.siu.edu/Raj%20Workshop%20Exercises%20with%20Tensorflow%20Playground.pdf

Machine Learning Crash Course – A self-study Guide

← → ⌂ 🔒 developers.google.com/machine-learning/crash-course

Machine Learning Crash Course Courses Practice Guides Glossary ▾

Search English ▾

Crash Course Problem Framing Data Prep Clustering Recommendation Testing and Debugging GANs

Google is committed to advancing racial equity for Black communities. [See how.](#)

Machine Learning Crash Course
with TensorFlow APIs

Google's fast-paced, practical introduction to machine learning

[Start Crash Course](#) [View prerequisites](#)

A self-study guide for aspiring
machine learning practitioners