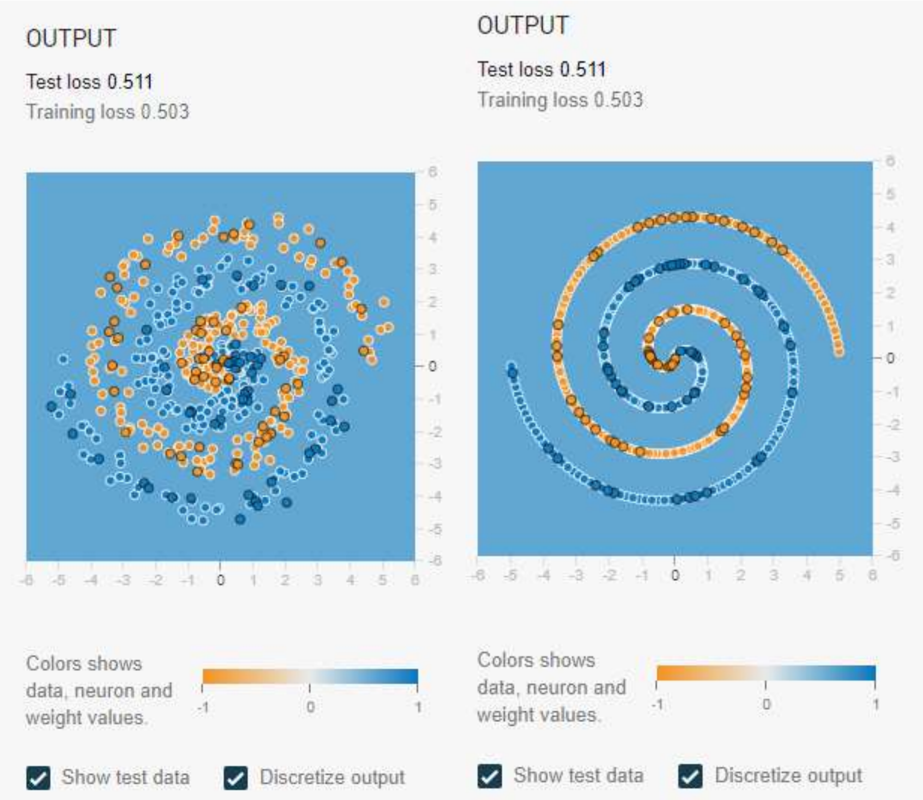
agenda

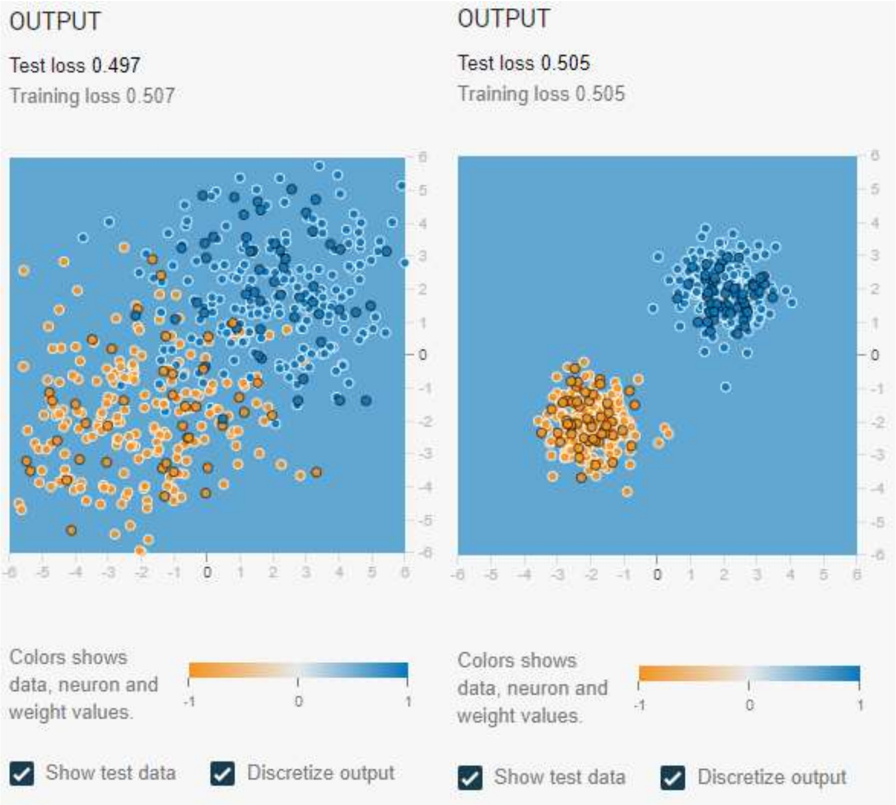
1. Toy problem to warming up
2. Review existing visualizations
3. Audio-video (multimodal)
   - Review existing visualizations
   - My work in progress.

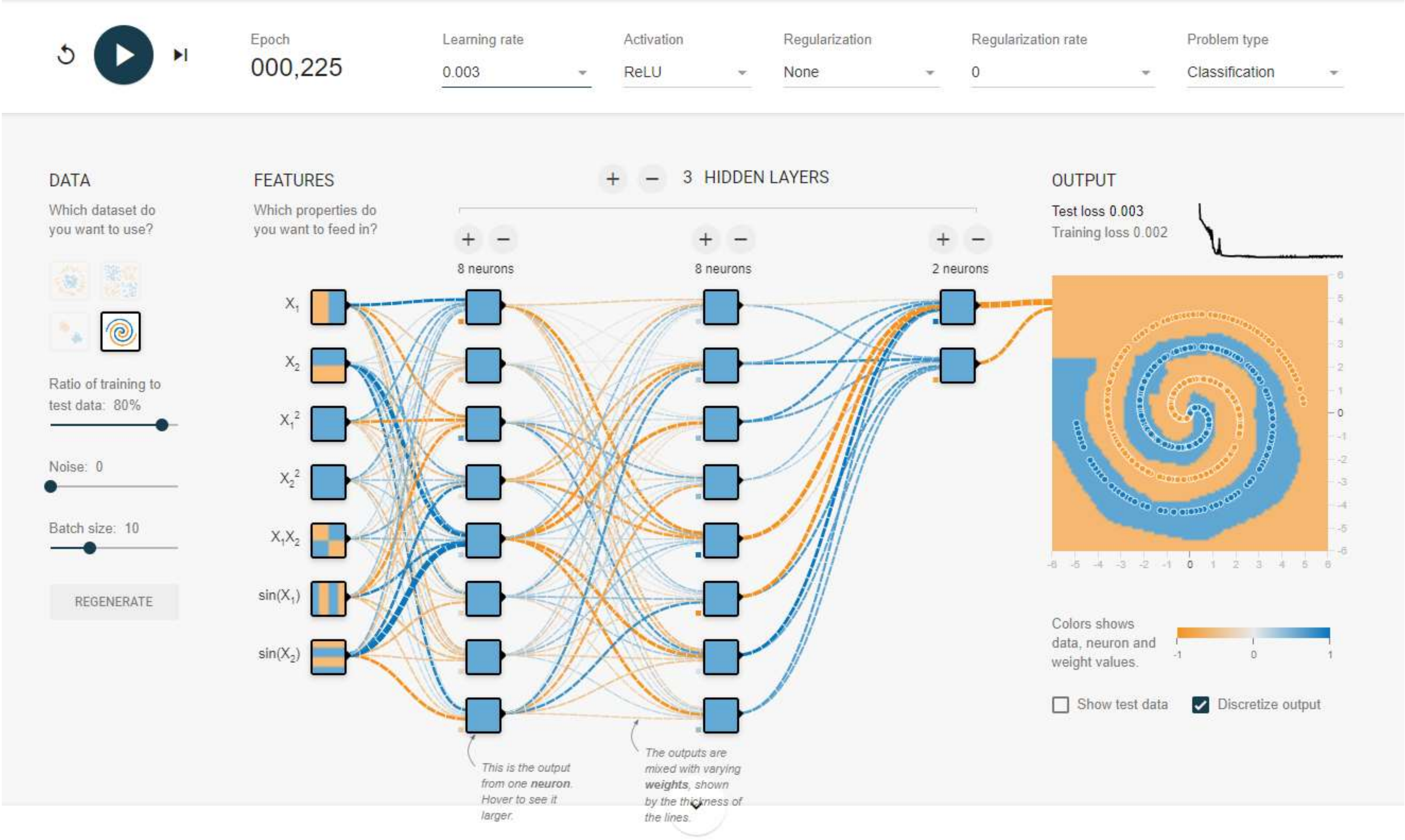

THIS IS YOUR MACHINE LEARNING SYSTEM?
YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.
WHAT IF THE ANSWERS ARE WRONG?
JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.
DATA

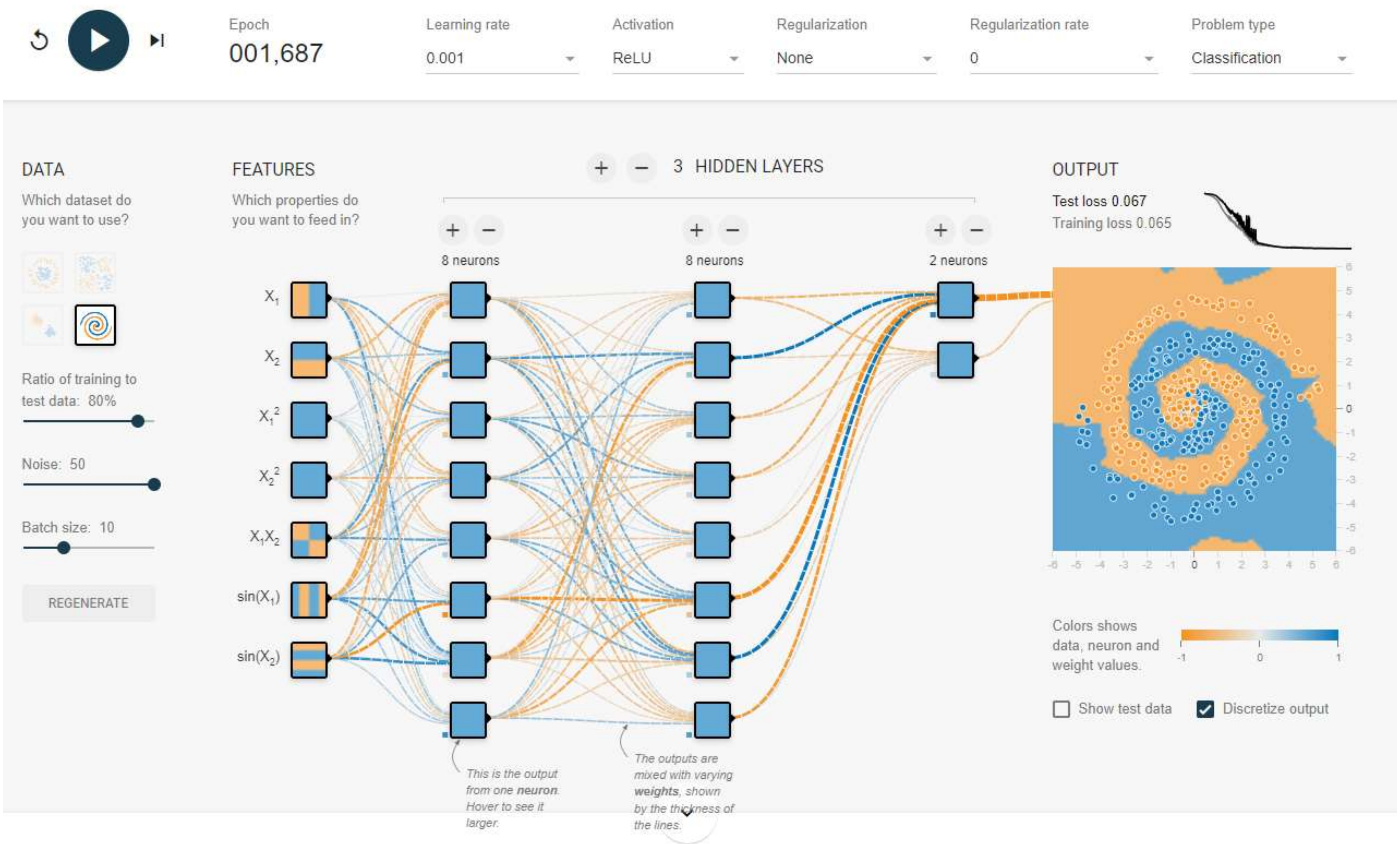
Toy problem to warming up

http://playground.tensorflow.org/

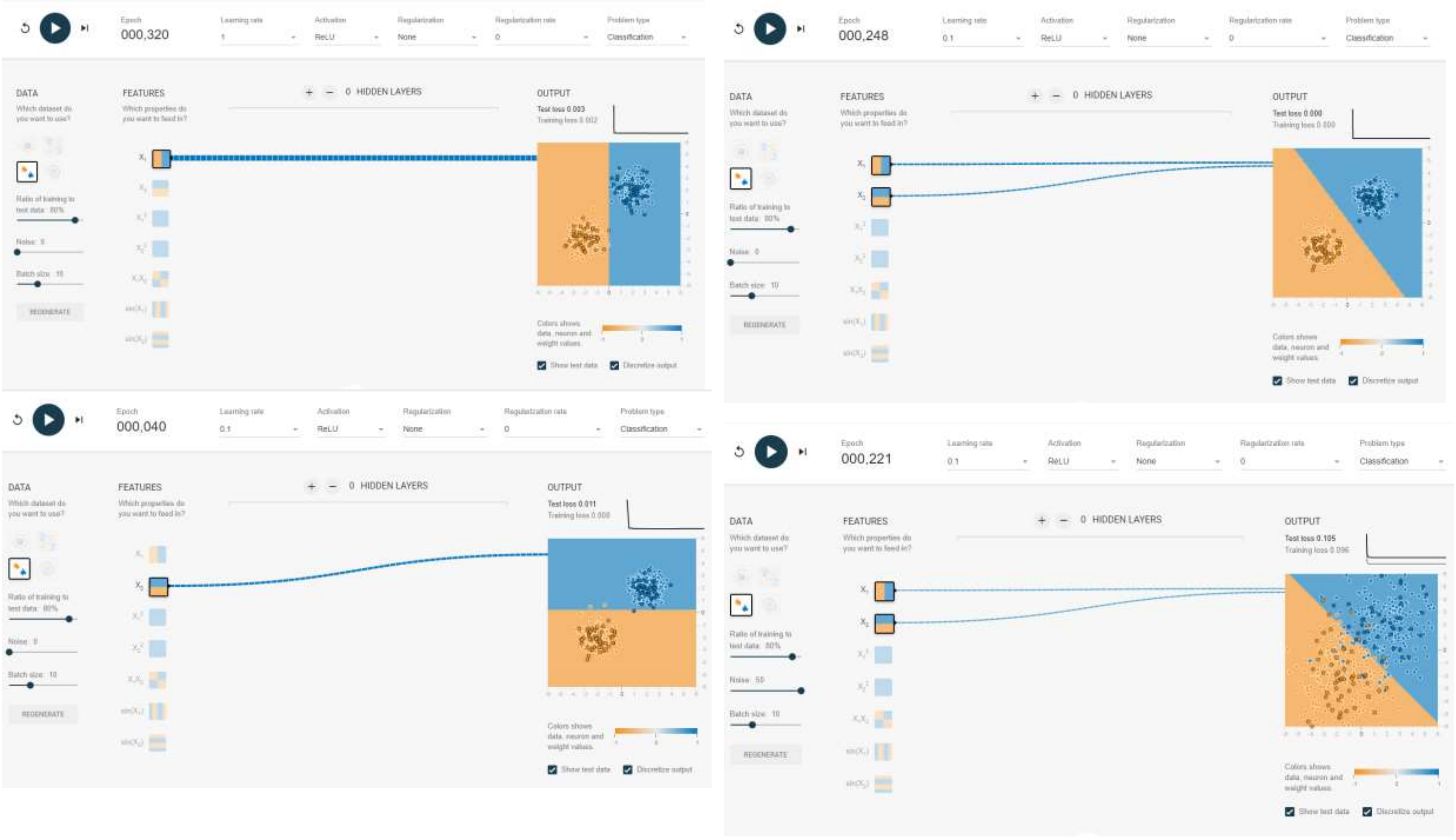


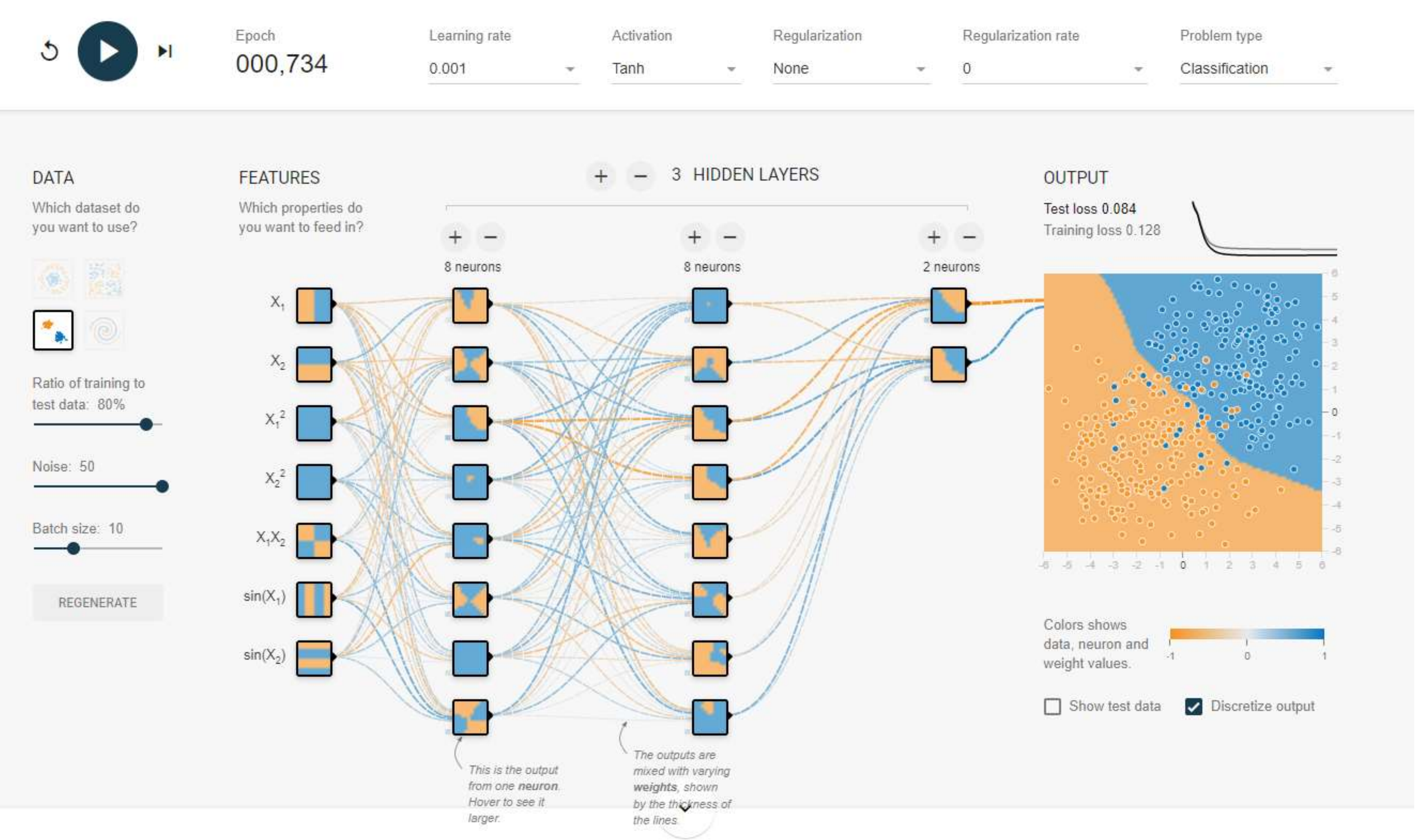


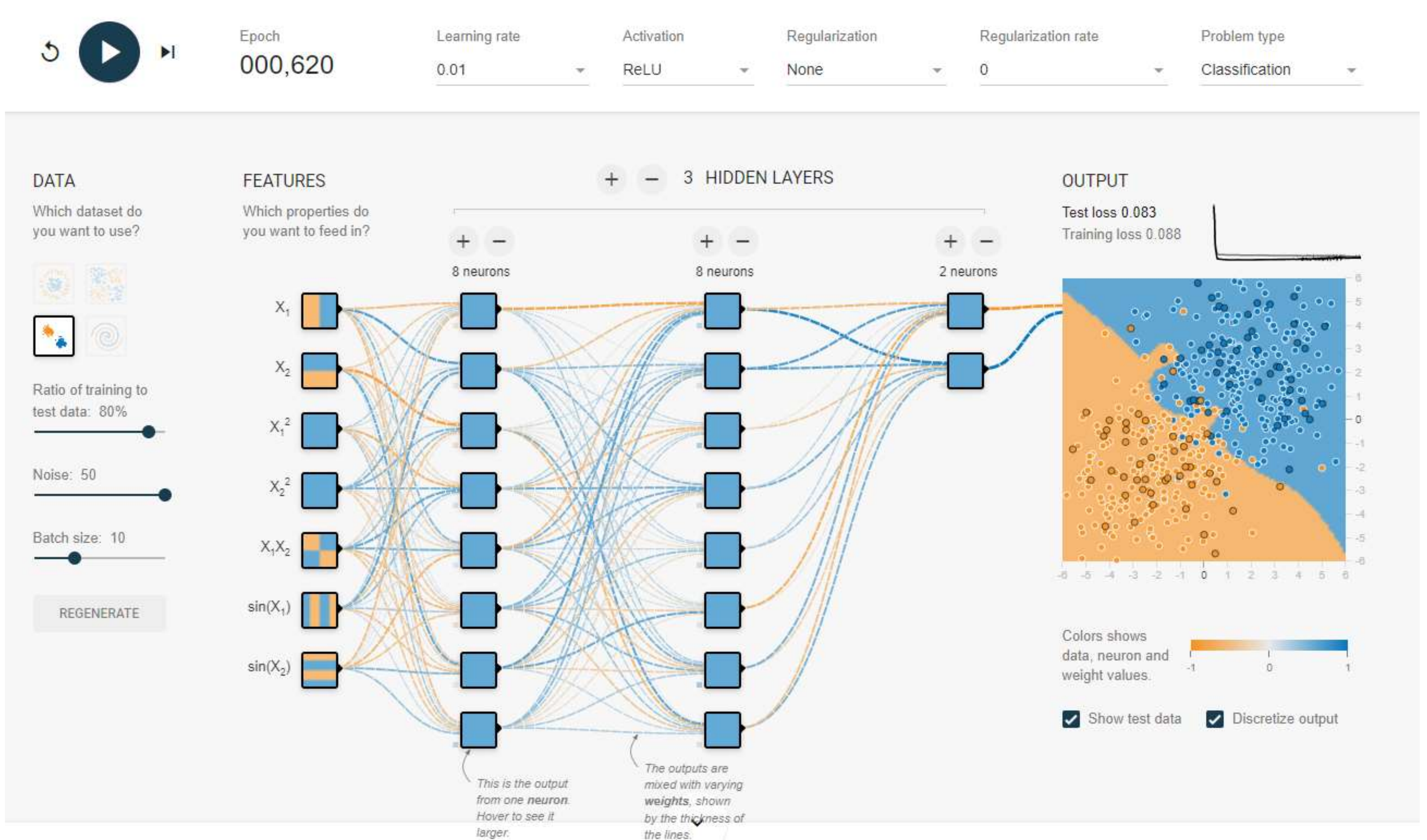

Epoch
000,225
Learning rate
0.003
Activation
ReLU
Regularization
None
Regularization rate
0
Problem type
Classification
DATA
Which dataset do you want to use?
Ratio of training to test data: 80%
Noise: 0
Batch size: 10
REGENERATE
FEATURES
Which properties do you want to feed in?
3 HIDDEN LAYERS
8 neurons
8 neurons
2 neurons
X_1
X_2
X_1^2
X_2^2
X_1X_2
$\sin(X_1)$
$\sin(X_2)$
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.
OUTPUT
Test loss 0.003
Training loss 0.002
Colors shows data, neuron and weight values.
Show test data
Discretize output



Epoch
001,687
Learning rate
0.001
Activation
ReLU
Regularization
None
Regularization rate
0
Problem type
Classification
DATA
Which dataset do you want to use?
Ratio of training to test data: 80%
Noise: 50
Batch size: 10
REGENERATE
FEATURES
Which properties do you want to feed in?
3 HIDDEN LAYERS
8 neurons
8 neurons
2 neurons
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.
OUTPUT
Test loss 0.067
Training loss 0.065
Colors shows data, neuron and weight values.
Show test data
Discretize output


Epoch
000,320
Learning rate
1
Activation
ReLU
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?
0 HIDDEN LAYERS
OUTPUT
Test loss 0.003
Training loss 0.002
REGENERATE
Show test data
Discretize output
Epoch
000,248
Learning rate
0.1
Activation
ReLU
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?
0 HIDDEN LAYERS
OUTPUT
Test loss 0.000
Training loss 0.000
Ratio of training to test data: 80%
Noise: 0
Batch size: 10
REGENERATE
Colors shows data, neuron and weight values.
Show test data
Discretize output
Epoch
000,040
Learning rate
0.1
Activation
ReLU
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?
0 HIDDEN LAYERS
OUTPUT
Test loss 0.011
Training loss 0.008
Ratio of training to test data: 80%
Noise: 0
Batch size: 10
REGENERATE
Colors shows data, neuron and weight values.
Show test data
Discretize output
Epoch
000,221
Learning rate
0.1
Activation
ReLU
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?
0 HIDDEN LAYERS
OUTPUT
Test loss 0.105
Training loss 0.096
Ratio of training to test data: 80%
Noise: 50
Batch size: 10
REGENERATE
Colors shows data, neuron and weight values.
Show test data
Discretize output


Epoch
000,734
Learning rate
0.001
Activation
Tanh
Regularization
None
Regularization rate
0
Problem type
Classification
DATA
Which dataset do you want to use?
Ratio of training to test data: 80%
Noise: 50
Batch size: 10
REGENERATE
FEATURES
Which properties do you want to feed in?
X_1
X_2
X_1^2
X_2^2
X_1X_2
$\sin(X_1)$
$\sin(X_2)$
3 HIDDEN LAYERS
8 neurons
8 neurons
2 neurons
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.
OUTPUT
Test loss 0.084
Training loss 0.128
Colors shows data, neuron and weight values.
-1
0
1
Show test data
Discretize output


Epoch
000,620
Learning rate
0.01
Activation
ReLU
Regularization
None
Regularization rate
0
Problem type
Classification
DATA
Which dataset do you want to use?
Ratio of training to test data: 80%
Noise: 50
Batch size: 10
REGENERATE
FEATURES
Which properties do you want to feed in?
X1
X2
X1²
X2²
X1X2
sin(X1)
sin(X2)
3 HIDDEN LAYERS
8 neurons
8 neurons
2 neurons
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.
OUTPUT
Test loss 0.083
Training loss 0.088
Colors shows data, neuron and weight values.
-1
0
1
Show test data
Discretize output

- Seems simpler model the better
- If know what the answer should be, maybe you do not ask any more;)

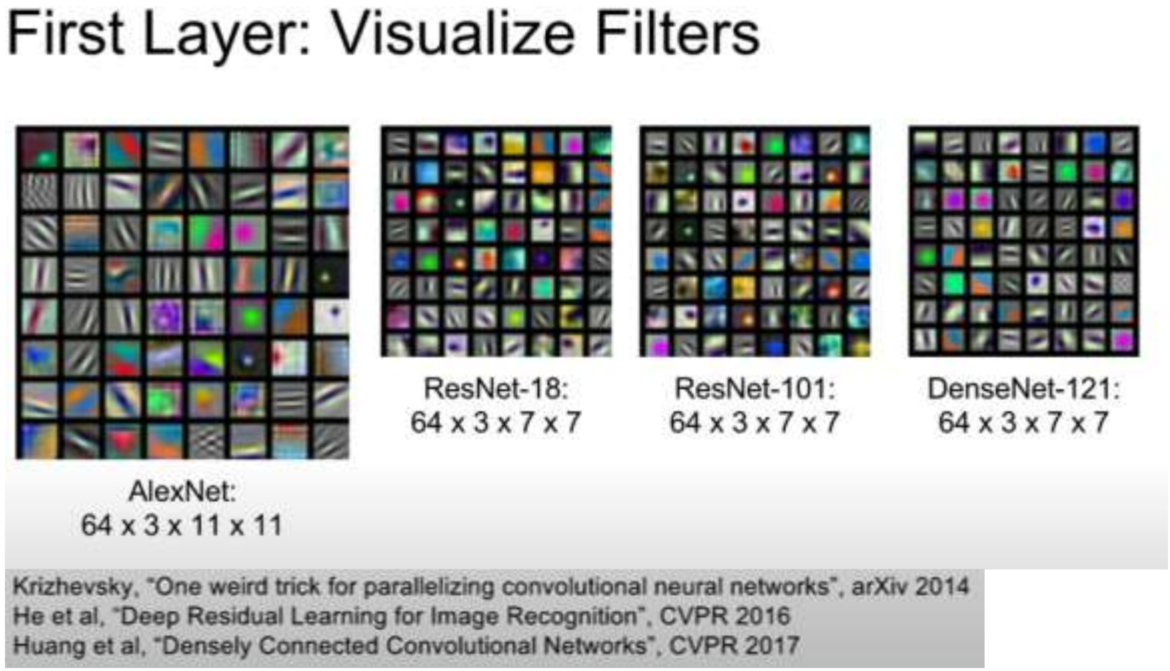
Short review of existing visualizations

- based on Lecture 12: Visualizing and Understanding, by Stanford University School of Engineering

- What is going inside model
- Feature importance analysis (which features, how important are for what class prediction, what part of input data is important for particular class prediction,…)
- I just wanted to get some ‘best’ XAI method and apply it for my model…

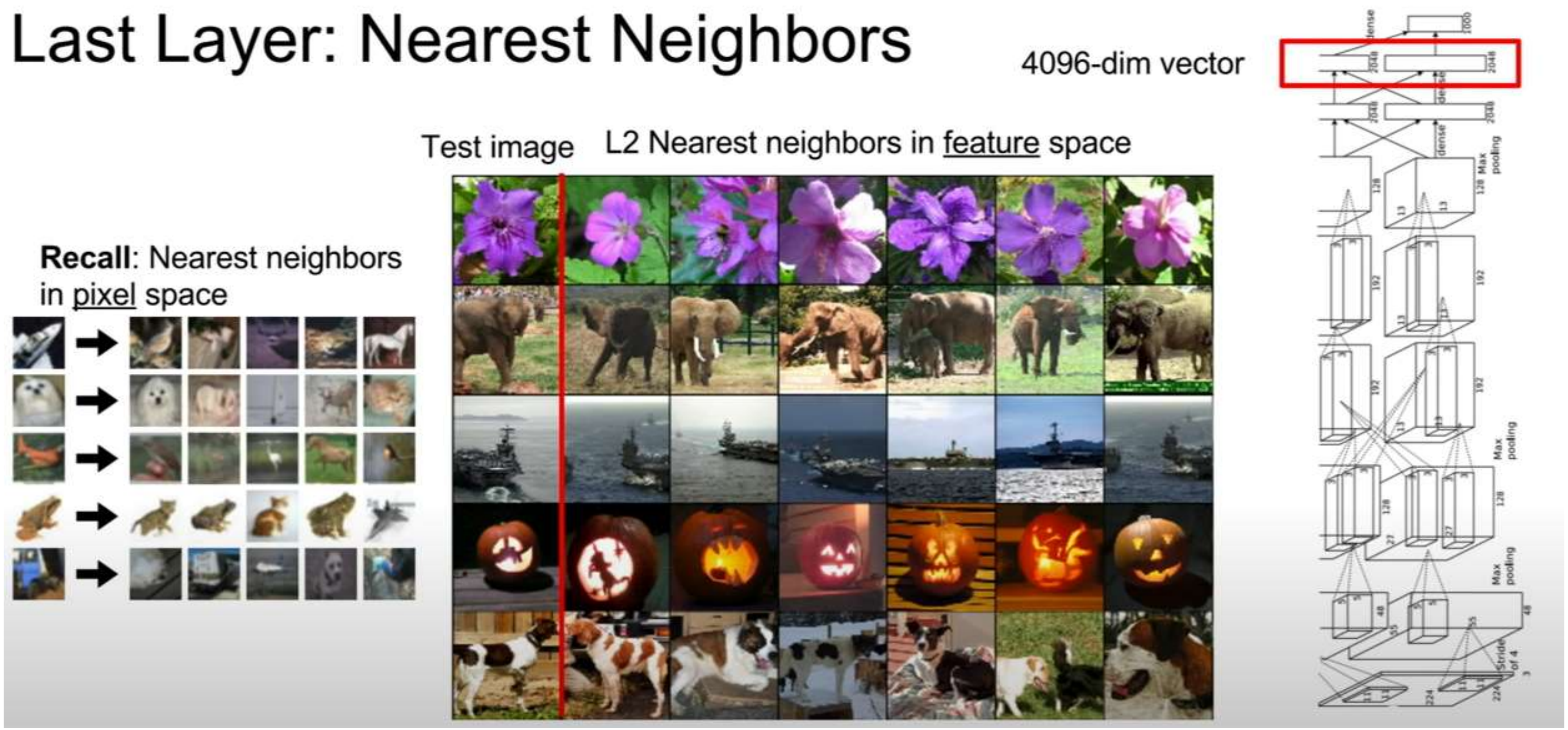
Short review of existing visualizations

First layer filters visualization – works for first layer, the deeper the more complex interpretation become (usually more filters, and more channels, also depend directly on previous layer after nonlinear activation not on original image)



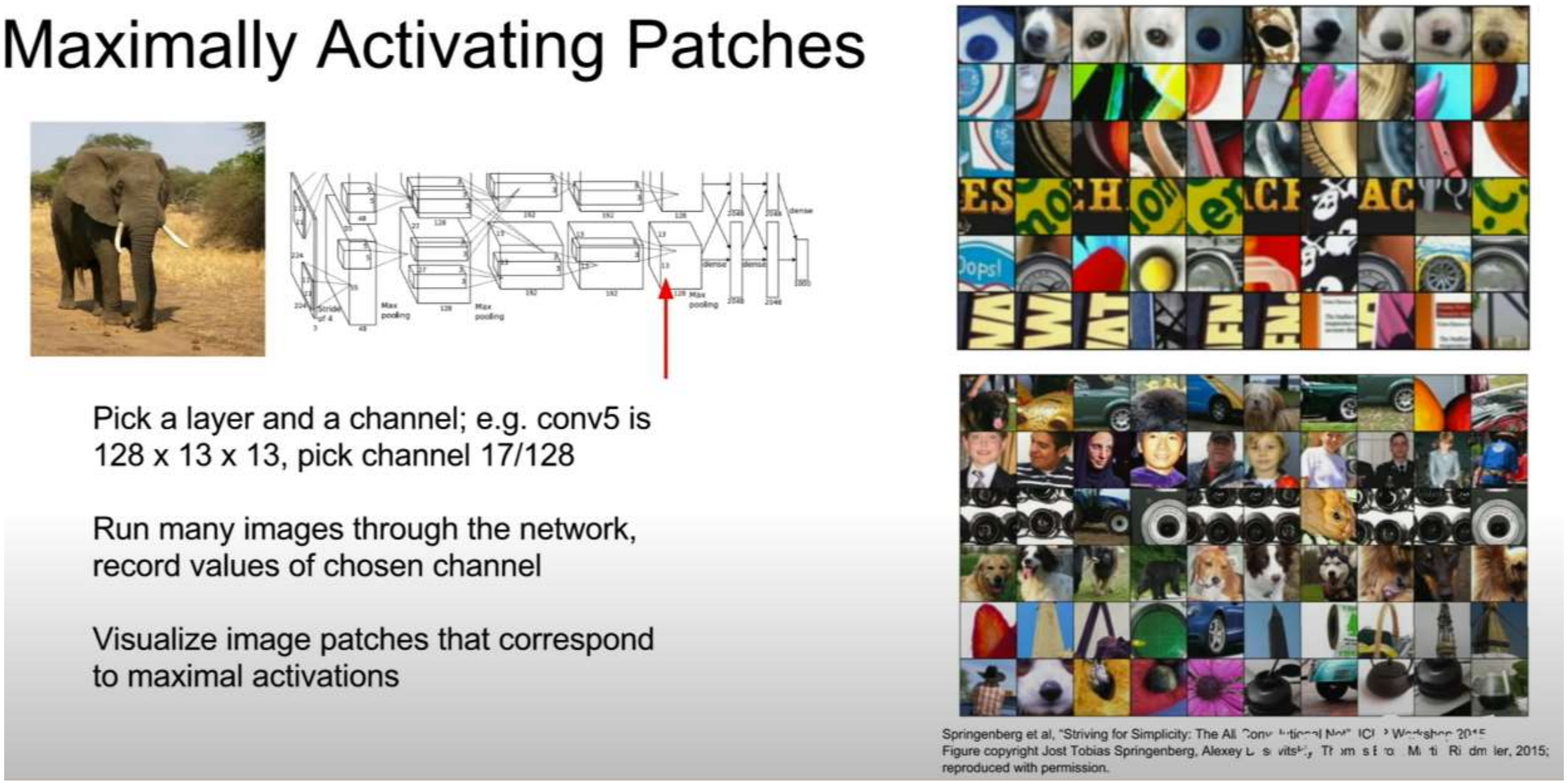
Meaning:
Visualizing filters means that filters looks on such patterns in input data, it is because scalar product of input data with filter is maximized once input data match filter.

Short review of existing visualizations



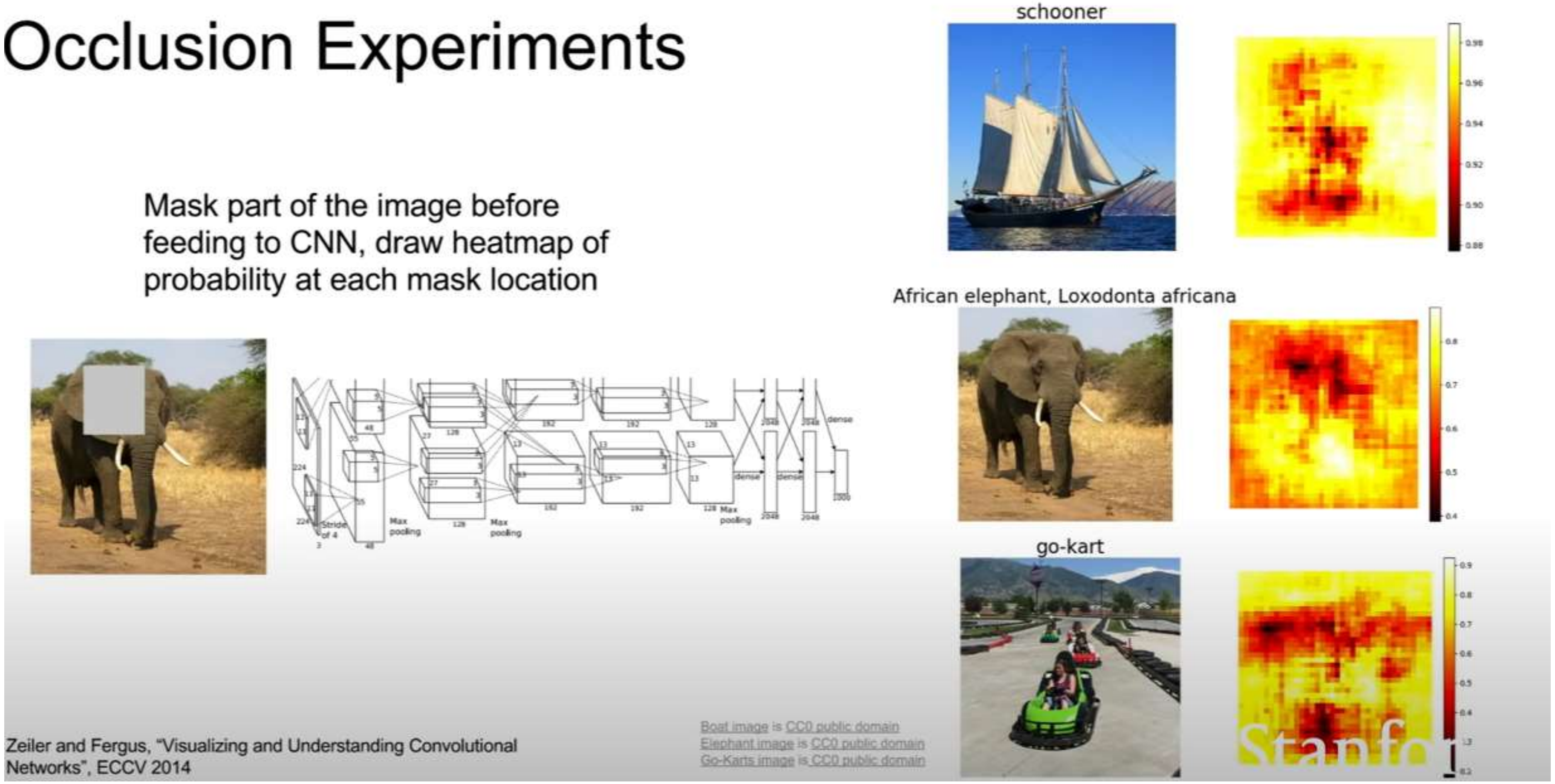
Van der Maaten and Hinton, "Visualizing Data using t-SNE", JMLR 2008
Krizhevsky et al, "ImageNet Classification with Deep Convolutional Neural Networks", NIPS 2012.

Short review of existing visualizations

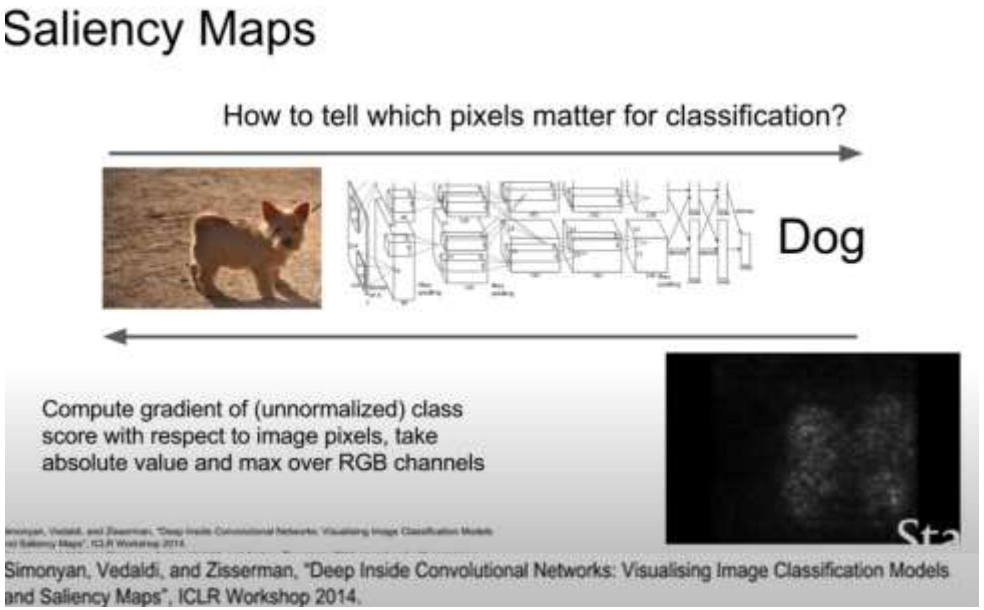


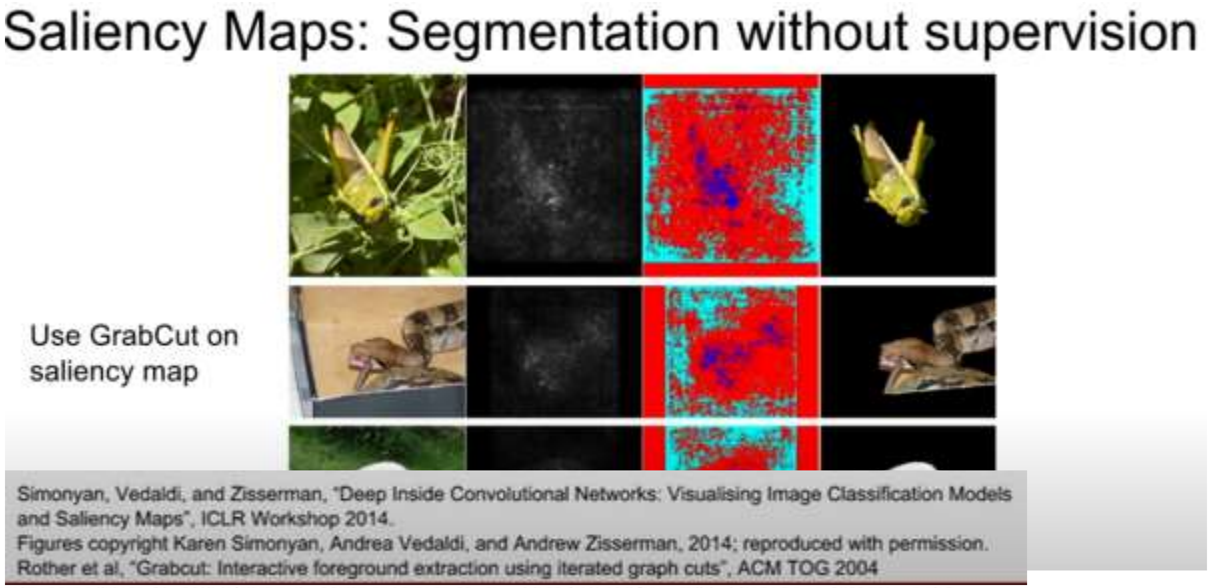
Springenberg, Striving for Simplicity: The All Convolutional Net, ICLR Workshop 2014

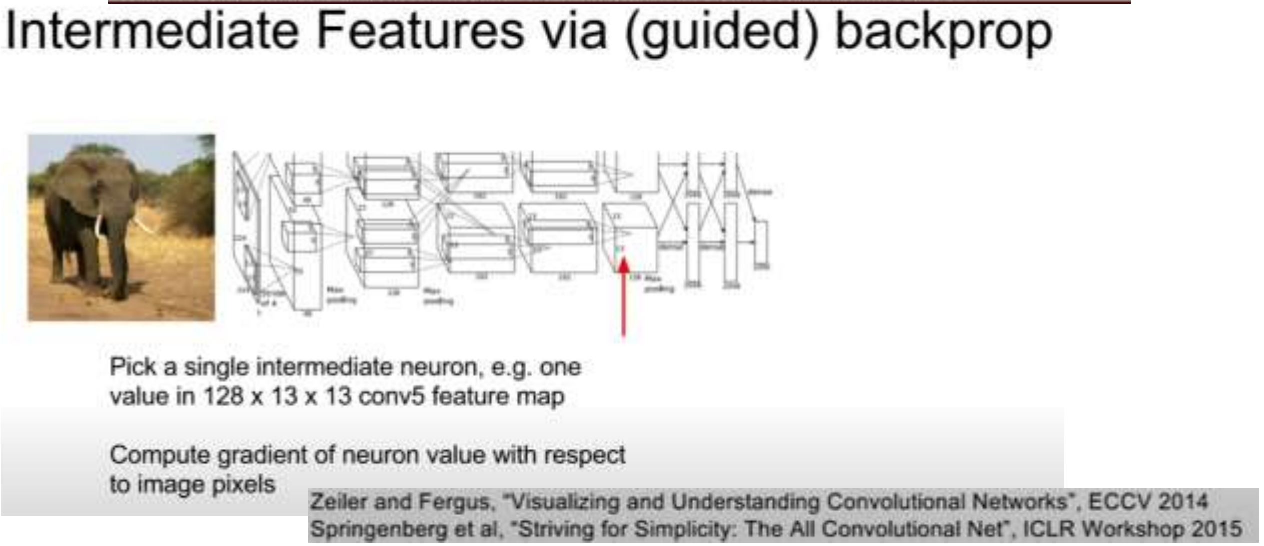
Short review of existing visualizations



Short review of existing visualizations







Related to fixed input image

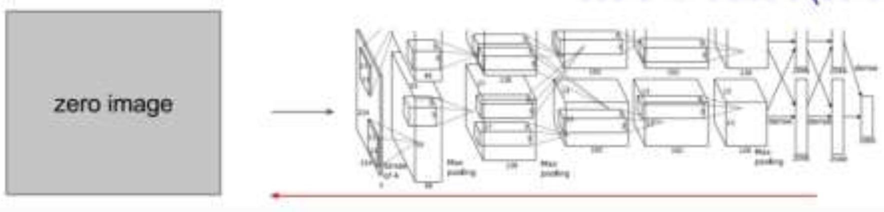
Short review of existing visualizations

Visualizing CNN features: Gradient Ascent

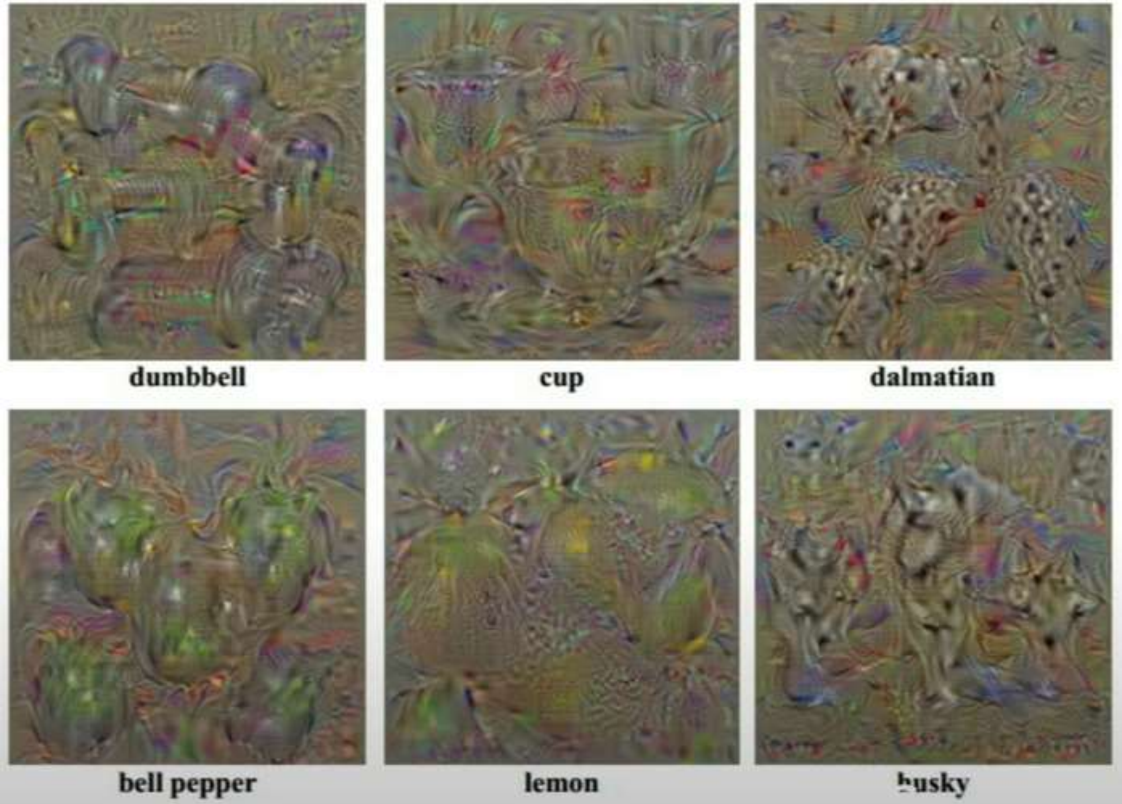
$$\arg\max_{I} \boxed{S_c(I)} - \lambda \|I\|_2^2$$

score for class c (before Softmax)

1. Initialize image to zeros



Repeat:
2. Forward image to compute current scores
3. Backprop to get gradient of neuron value with respect to image pixels
4. Make a small update to the image



Simonyan, Vedaldi, and Zisserman, "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps", ICLR Workshop 2014.

Yosinski et al, "Understanding Neural Networks Through Deep Visualization", ICML DL Workshop 2014.

Nguyen et al, "Synthesizing the preferred inputs for neurons in neural networks via deep generator networks," NIPS 2016

Mordvintsev et al. Inceptionism: Going Deeper into Neural Networks, 2015

Audio – video visualization

1. Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. arXiv preprint arXiv:1804.03619, 2018.

2. Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. The conversation: Deep audio-visual speech enhancement. In Interspeech, pages 3244–3248, 2018

3. Triantafyllos Afouras, Andrew Owens, Joon Son Chung, and Andrew Zisserman. Self-supervised learning of audio-visual objects from video. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16, pages 208–224. Springer, 2020

4. Ruohan Gao and Kristen Grauman. Co-separating sounds of visual objects. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pages 3878–3887, 2019

5. Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 1735–1744, 2019

6. Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, Computer Vision – ECCV 2018,

7. Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. My lips are concealed: Audio-visual speech enhancement through obstructions. arXiv preprint arXiv:1907.04975, 2019

8. Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In Proceedings of the European Conference on Computer Vision (ECCV), pages 631–648, 2018

Audio – video visualizations

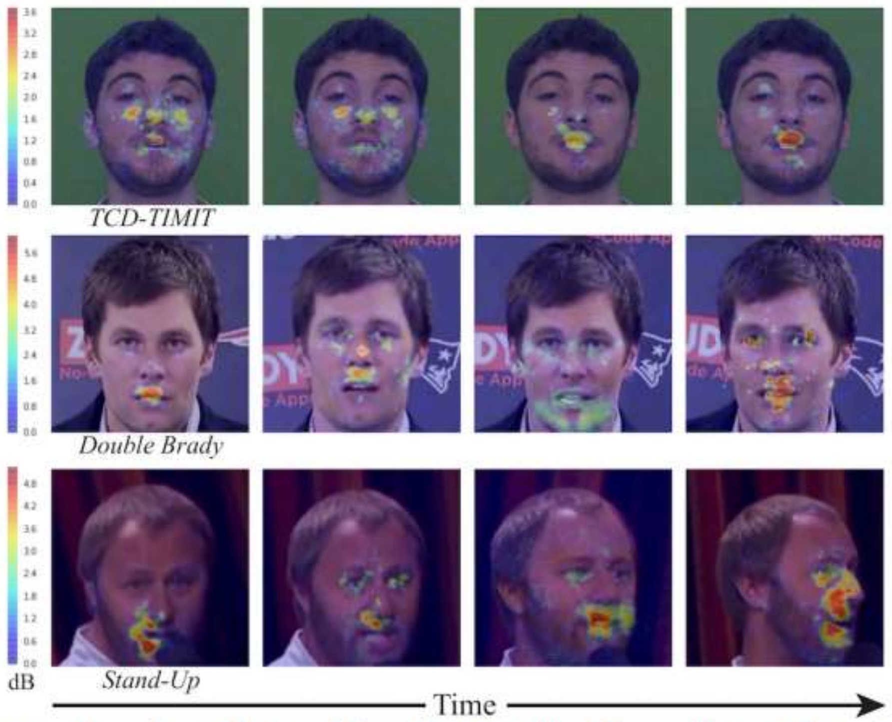
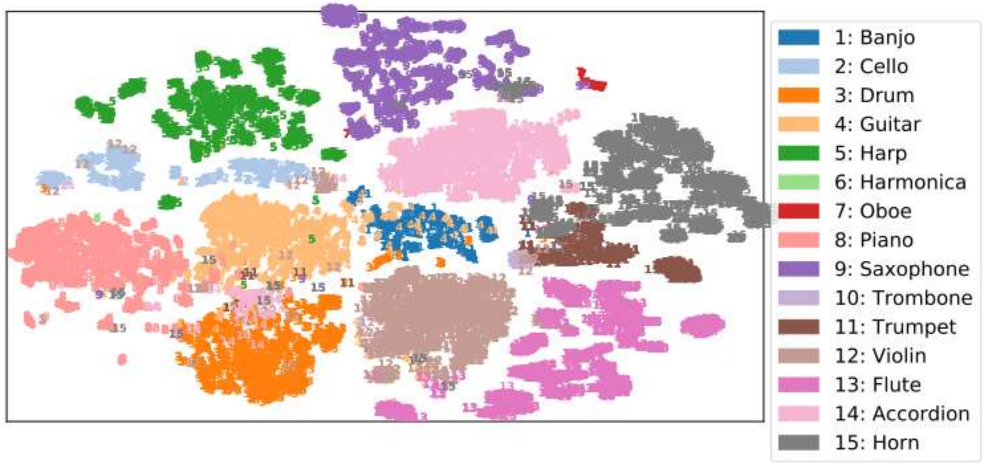


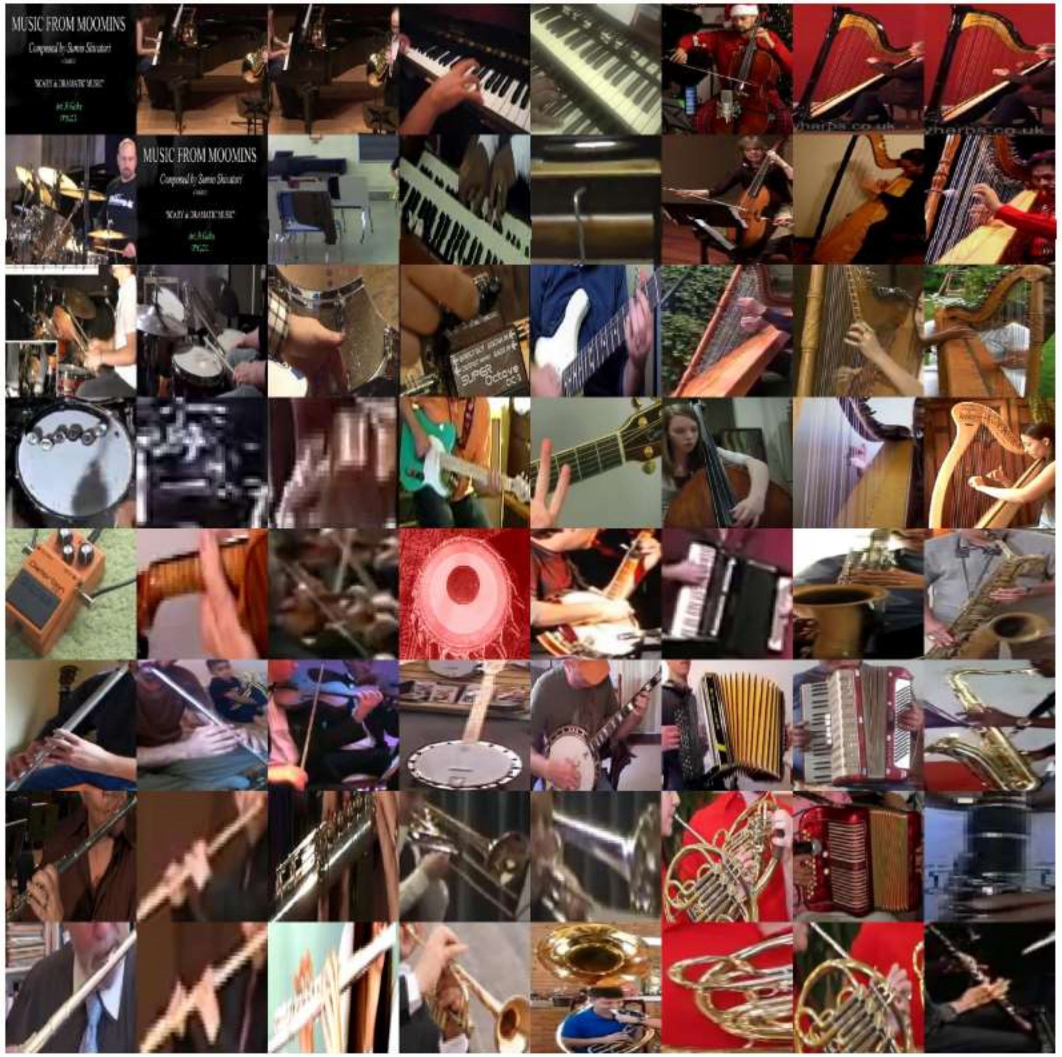
Fig. 8. **How does the model utilize the visual signal?** We show heat maps overlaid on representative input frames from several videos, visualizing the contribution of different regions of the frames to our speech separation result (in dB, see text), from blue (low contribution) to red (high contribution).

Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. arXiv preprint arXiv:1804.03619, 2018.

Audio – video visualizations



Embedding of separated sounds in AudioSet visualized with t-SNE in two ways: (top) categories are color-coded, and (left) visual objects are shown at their sound's embedding.



Ruohan Gao and Kristen Grauman. Co-separating sounds of visual objects. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pages 3878–3887, 2019

Audio – video visualizations

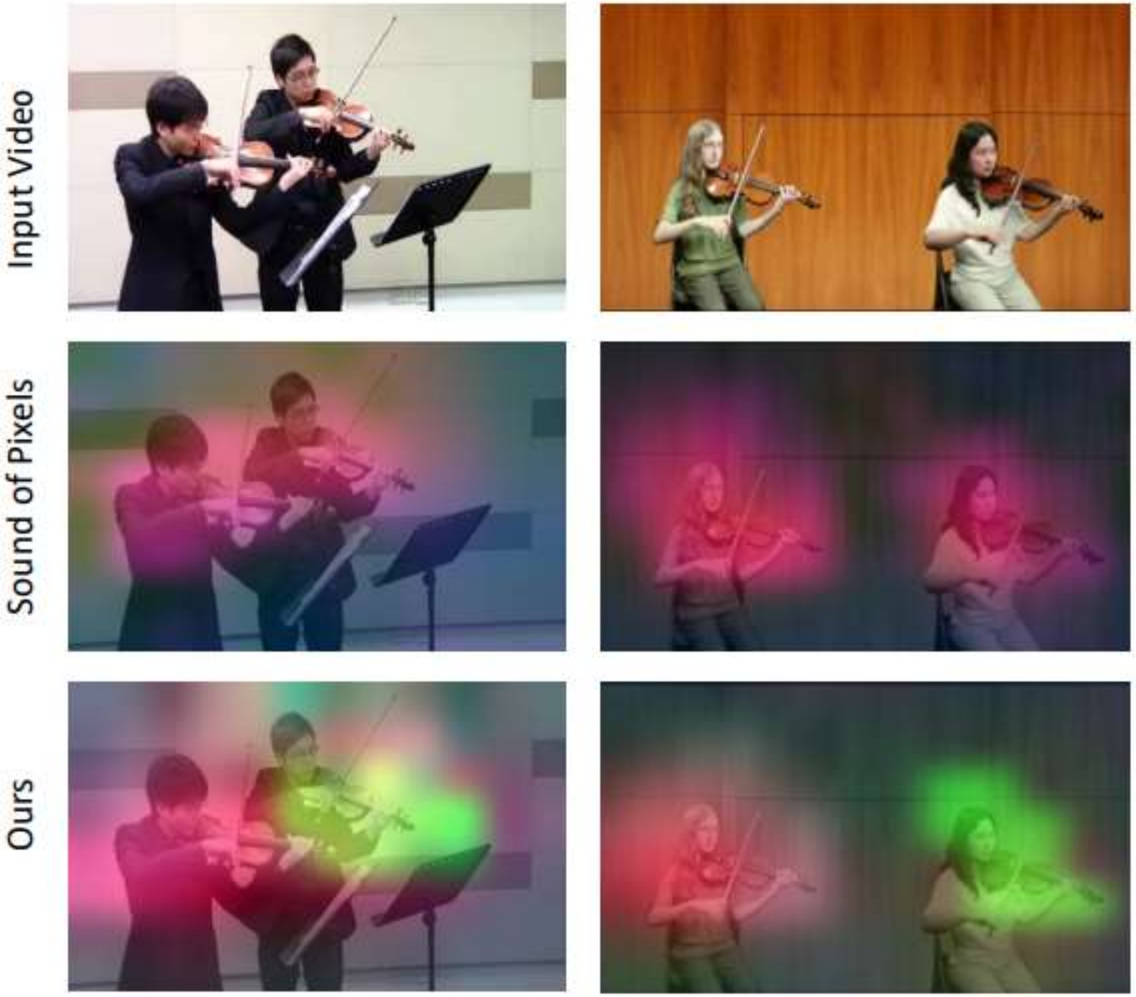


Figure 5. Pixel-level sound embedding results. To visualize the pixel-level sound separation results, we project sound features into a low dimensional space, and visualize them in RGB space. Different colors mean different sounds. Our model can tell the difference from duets of the same instruments, while Sound of Pixels model cannot.

We project sound features (vectorized spectrogram values) into a 3 dimensional space using PCA, and visualize them in color. Different colors in the heatmaps refer to different sounds. We show that our model can tell the difference from duets of the same instruments, while Sound of Pixels model cannot.

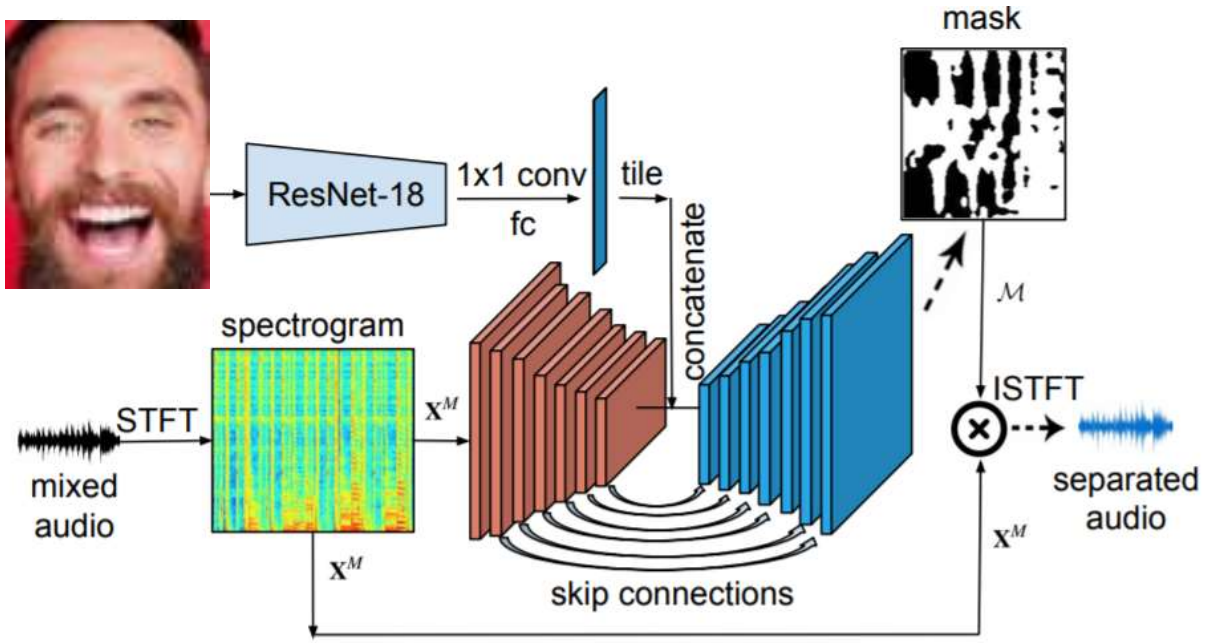
Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 1735–1744, 2019

Audio – video visualizations

Others:

- Benchmark audio or video subnetworks with audio only or video only counterparts to show how good representation is, if is useful for tasks different than trained for.

My visualizations



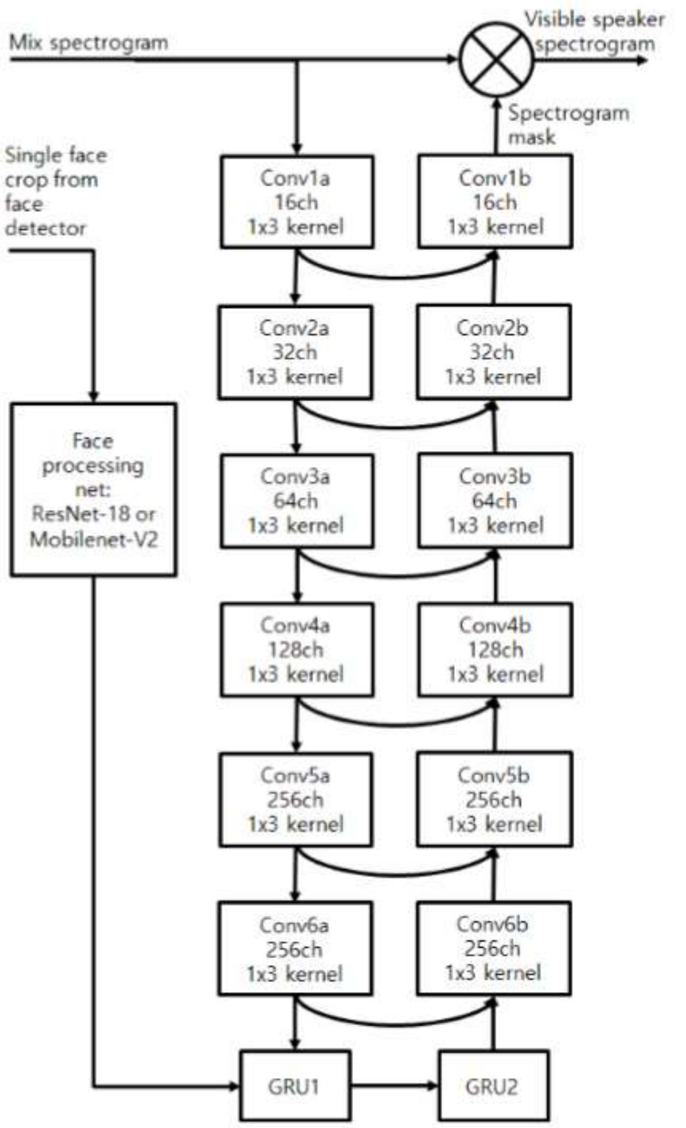
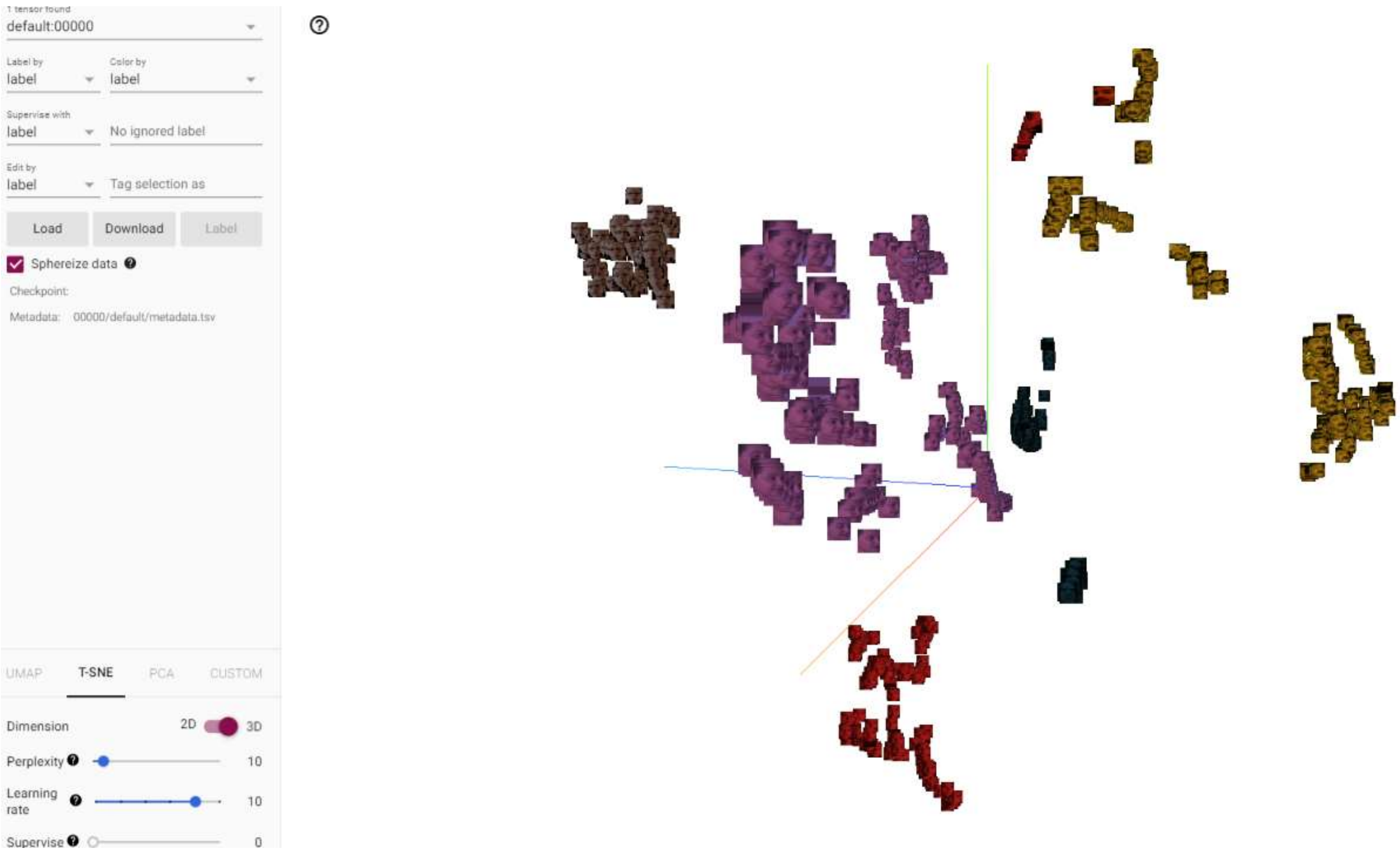


Figure 1. Audio-video model. It is model for speech enhancement [30] extended by adding video subnet covering face detector and face features extractor network (Resnet as baseline or Mobilenet as lightweight)

My visualizations



THANK YOU!!!!