

## Raport do zajęć nr 6

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

**Epoch** – za każdym razem gdy zostaną wykorzystane wszystkie dane treningowe w celu aktualizacji wag (przedstawiają siłę połączeń w neuronie) mija jeden epoch.

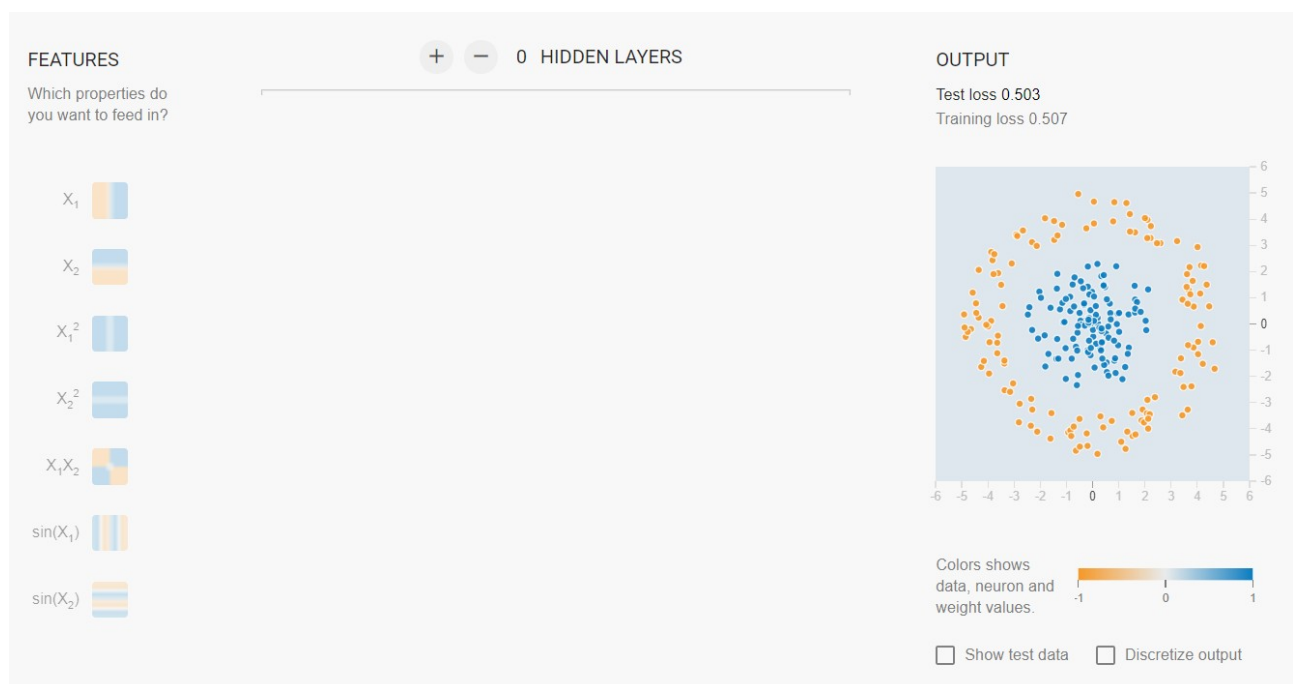
**Learning rate** – wskaźnik wpływający na szybkość uczenia się sieci neuronowej.

**Activation** – służy do wybierania funkcji aktywacyjnych, które są wykorzystywane gdy potrzebujemy nieliniowości.

**Regularization** – funkcja wykorzystywana do zapobiegania over fittingowi.

**Regularization rate** – określa wpływ regularyzacji.

**Problem type** – określa przekształcenia jakim będą poddawane dane.

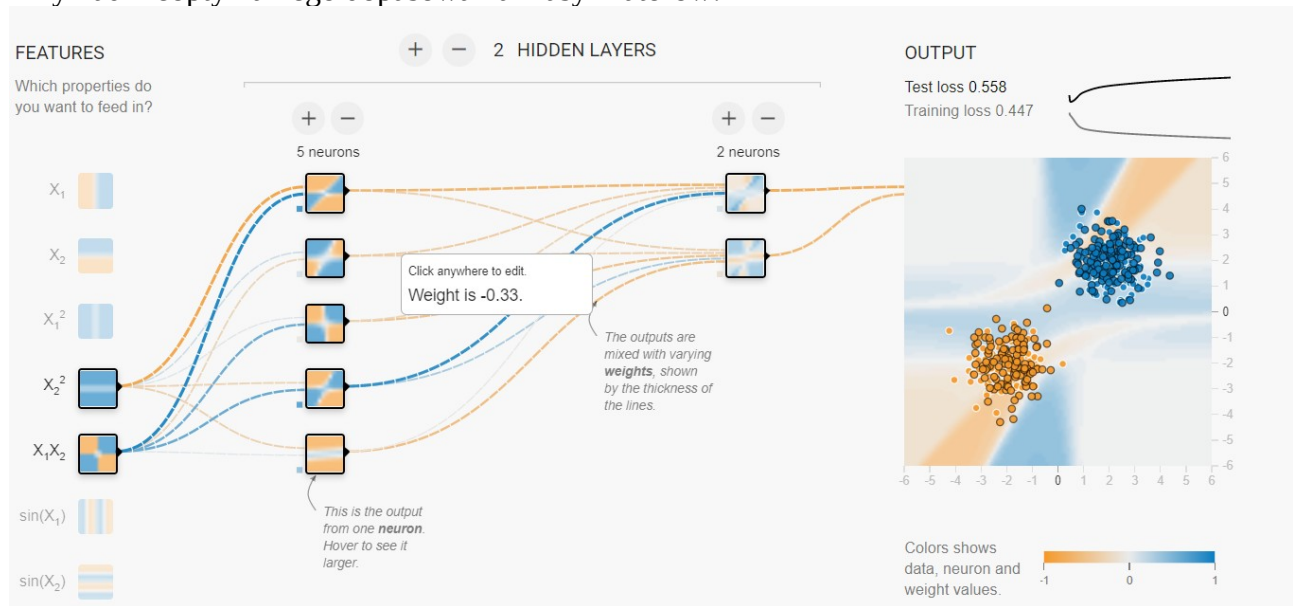


Features – wybór klasyfikatorów.

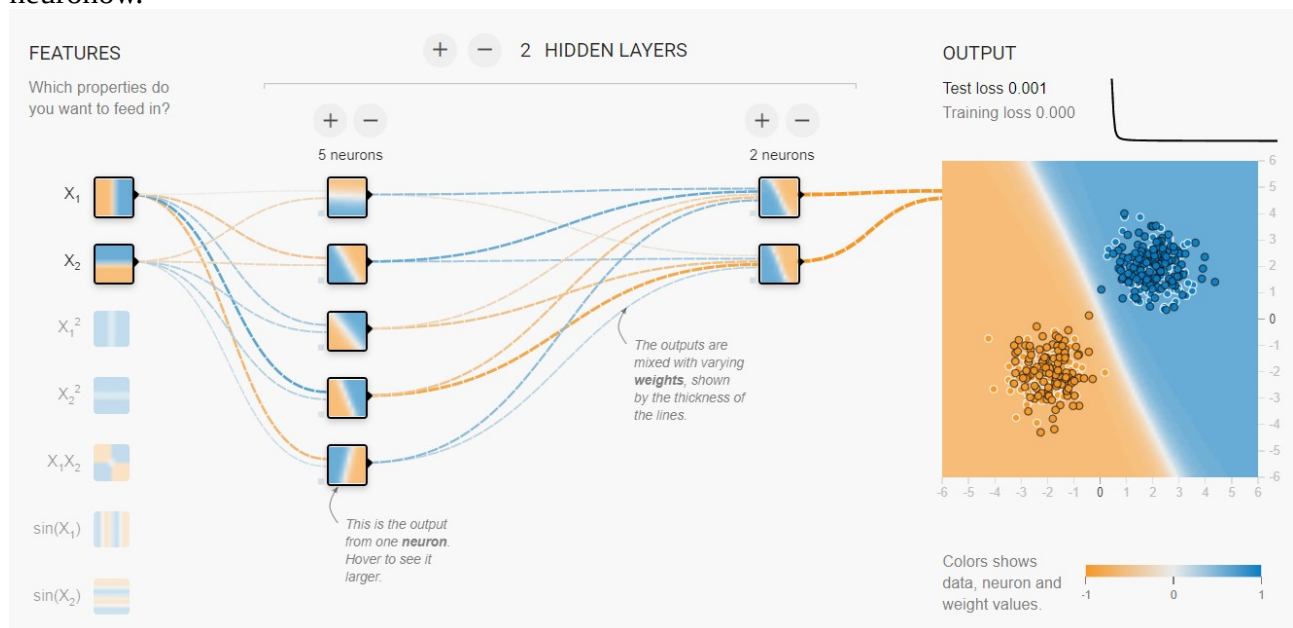
Hidden layers – Warstwy składające się z neuronów, które wykonują przekształcenia na danych wejściowych.

Output – wynik przekazywany przez neurony z warstwy wyjściowej, pokazuje również niedokładność wyniku testowego oraz treningowego.

## Przykład nieoptymalnego dopasowania klasyfikatorów:

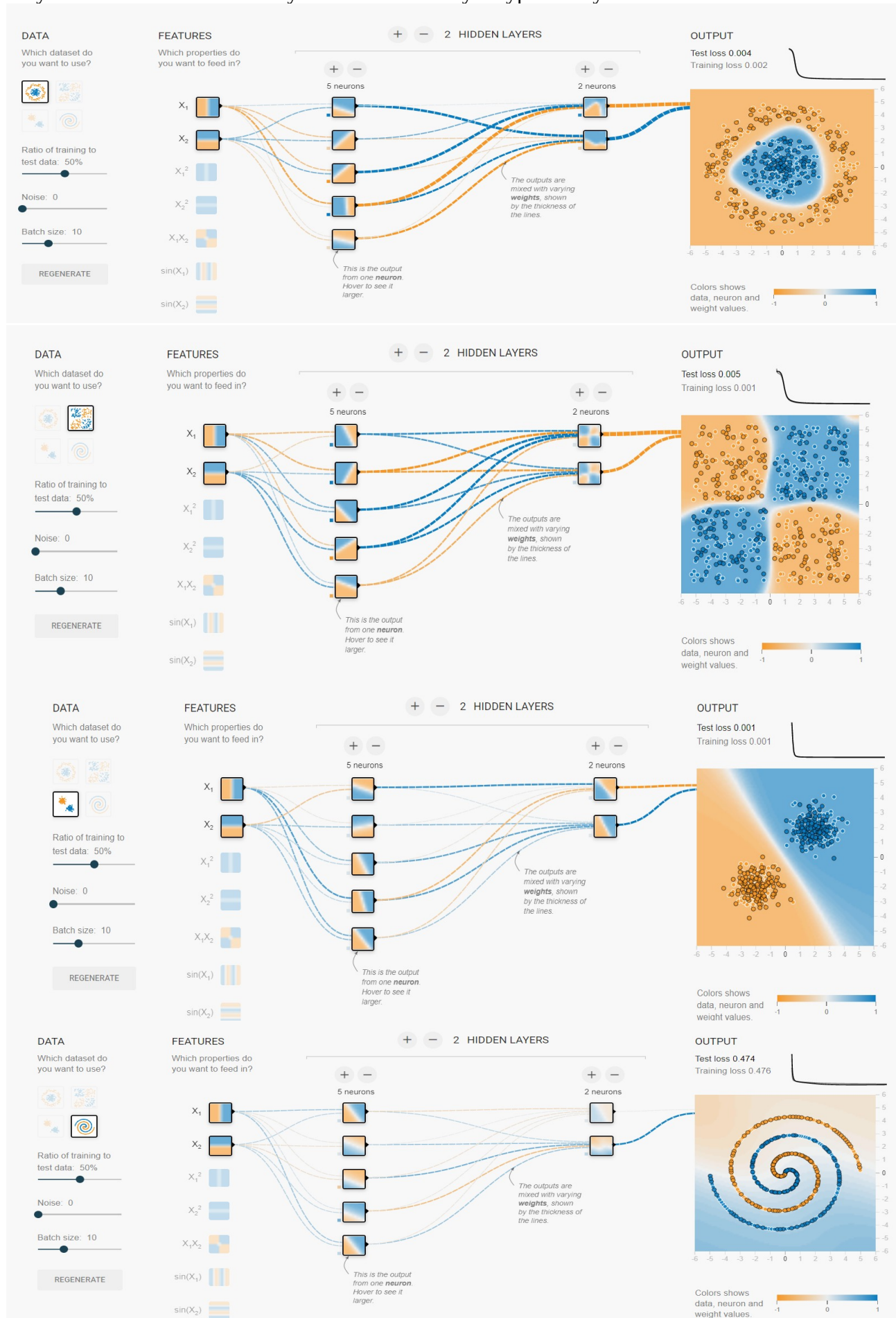


## Przykład poprawnego zastosowania klasyfikatorów dla tego samego zestawu danych i układu neuronów.



Zadanie klasyfikacji często może nie mieć możliwości wykonania przez niewłaściwy wybór klasyfikatorów.

## Przykładowe zastosowanie klasyfikatorów dla różnych typów danych:



#### DATA

Which dataset do you want to use?



Ratio of training to test data: 50%

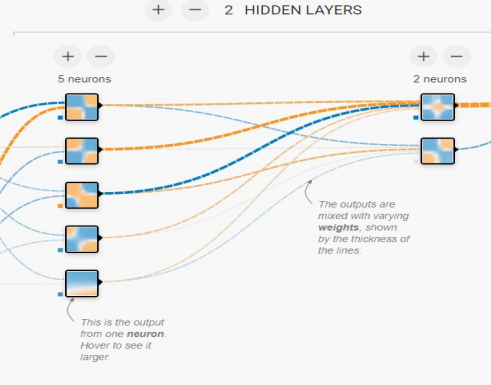
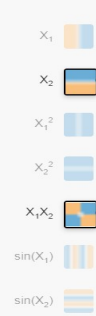
Noise: 0

Batch size: 10

REGENERATE

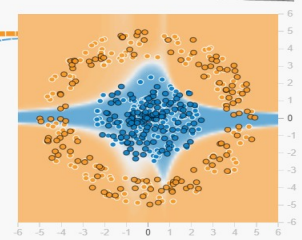
#### FEATURES

Which properties do you want to feed in?



#### OUTPUT

Test loss 0.124  
Training loss 0.068



Colors shows data, neuron and weight values.

☒ Show test data ☐ Discretize output

#### DATA

Which dataset do you want to use?



Ratio of training to test data: 50%

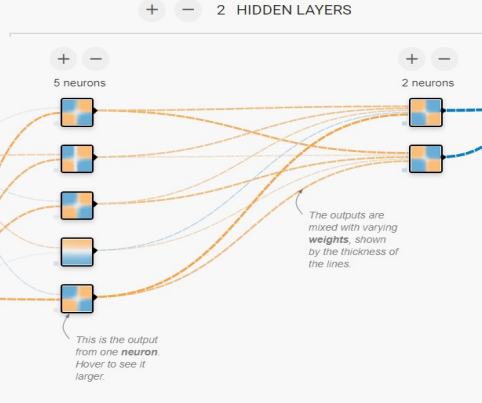
Noise: 0

Batch size: 10

REGENERATE

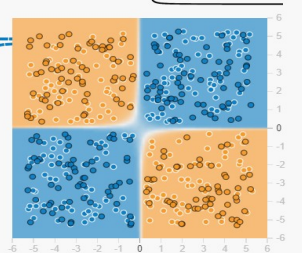
#### FEATURES

Which properties do you want to feed in?



#### OUTPUT

Test loss 0.000  
Training loss 0.001



Colors shows data, neuron and weight values.

☒ Show test data ☐ Discretize output

#### DATA

Which dataset do you want to use?



Ratio of training to test data: 50%

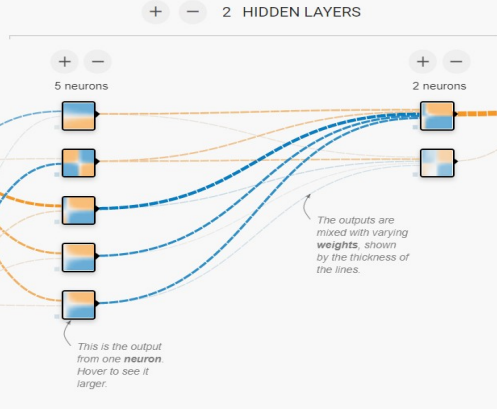
Noise: 0

Batch size: 10

REGENERATE

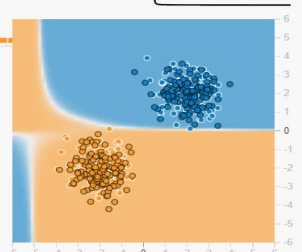
#### FEATURES

Which properties do you want to feed in?



#### OUTPUT

Test loss 0.000  
Training loss 0.001



Colors shows data, neuron and weight values.

☒ Show test data ☐ Discretize output

#### DATA

Which dataset do you want to use?



Ratio of training to test data: 50%

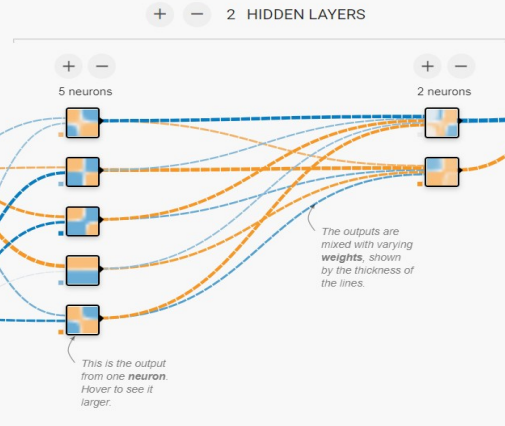
Noise: 0

Batch size: 10

REGENERATE

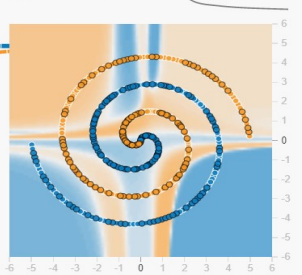
#### FEATURES

Which properties do you want to feed in?



#### OUTPUT

Test loss 0.484  
Training loss 0.329



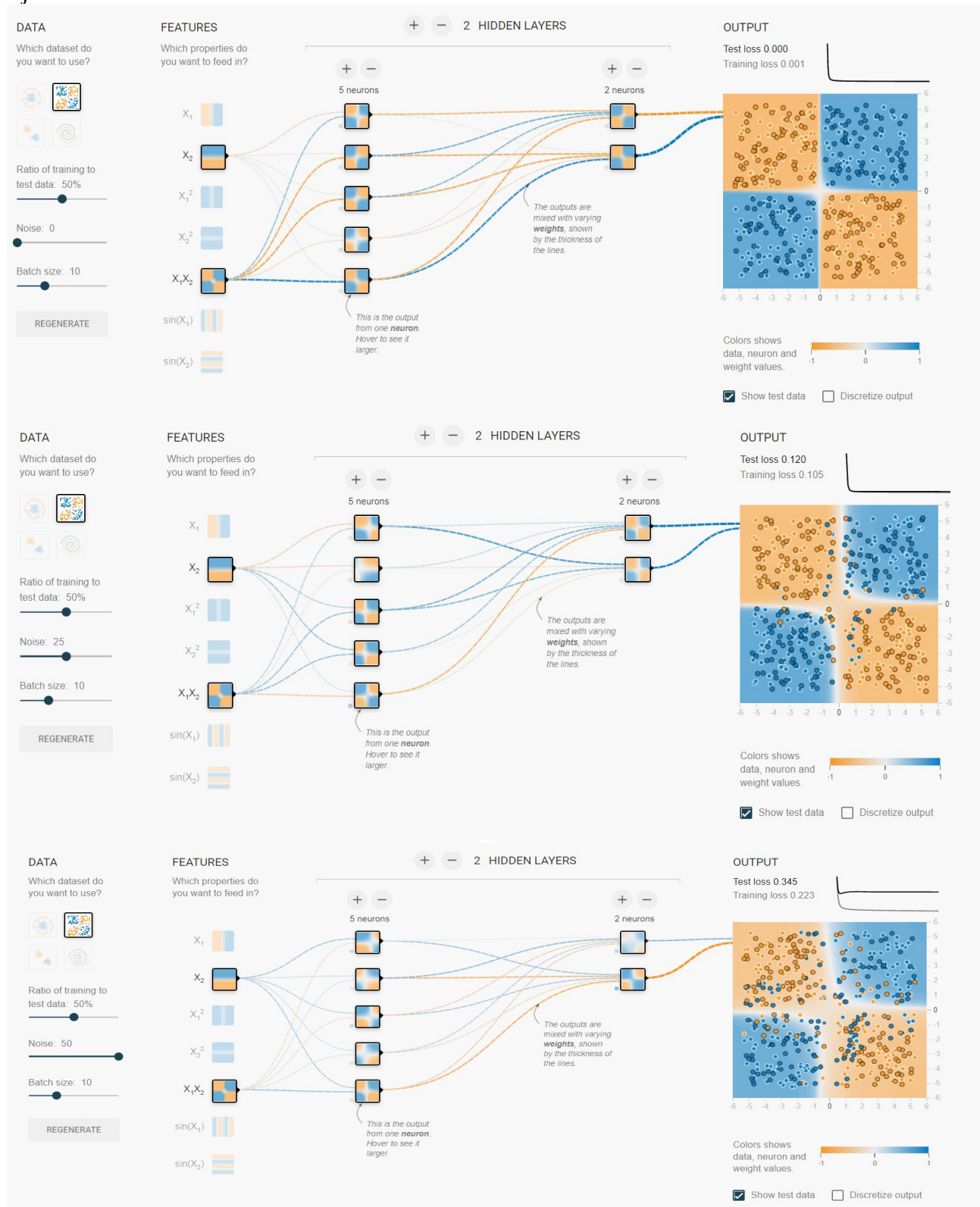
Colors shows data, neuron and weight values.

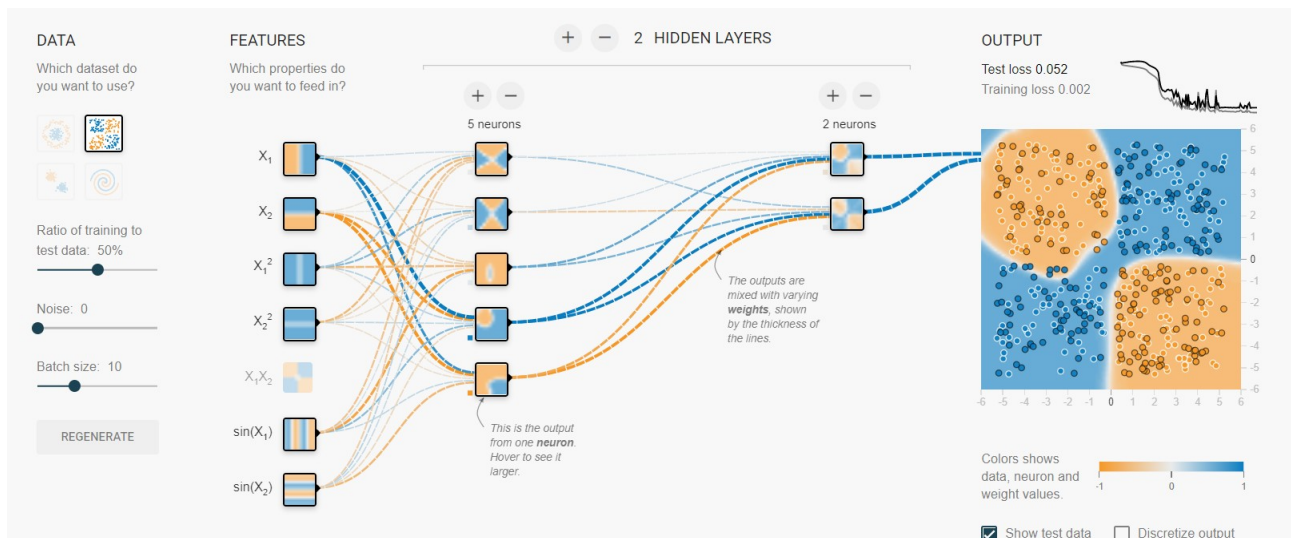
☒ Show test data ☐ Discretize output



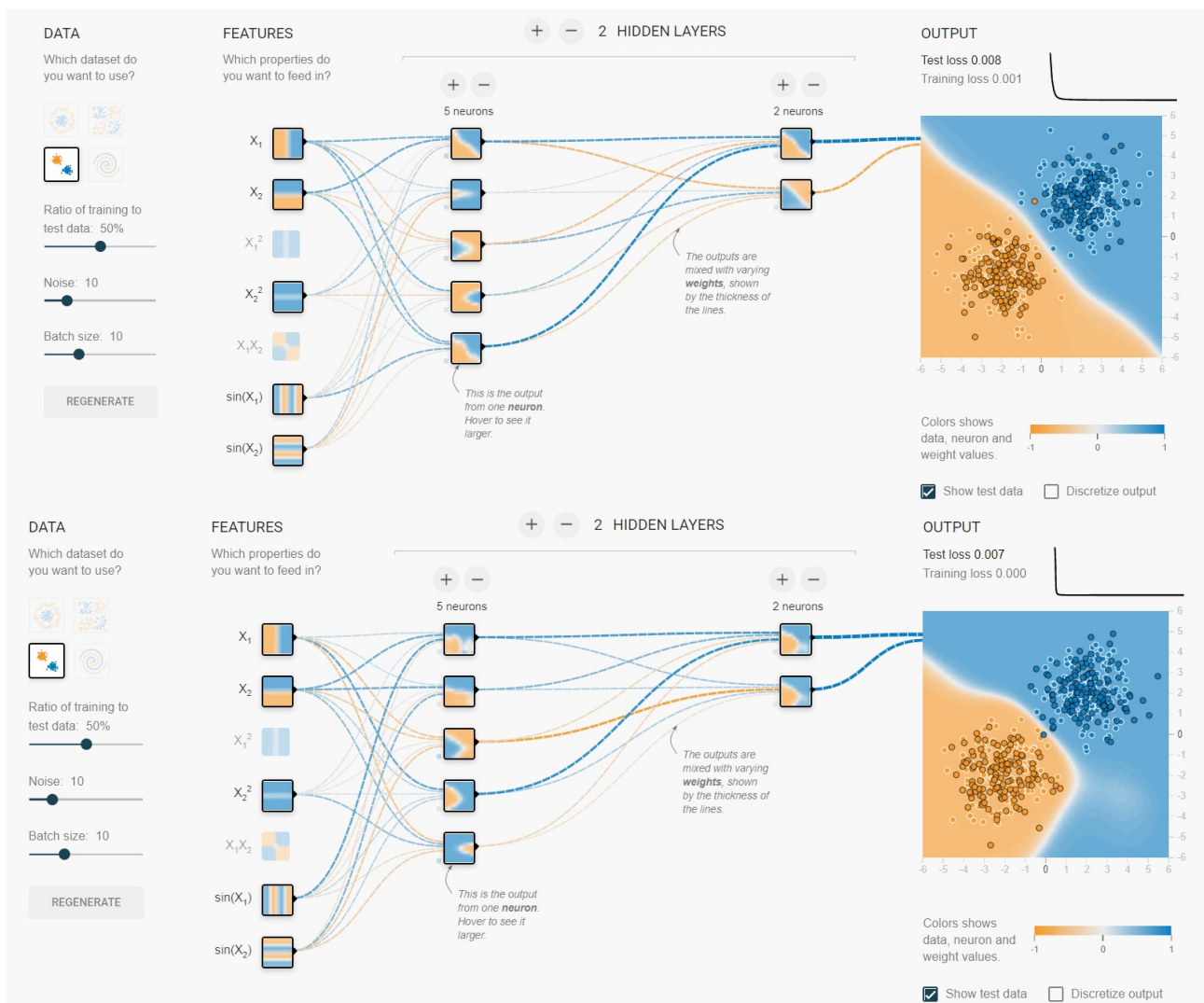
Pierwszy zestaw klasyfikatorów najlepiej nadaje się do pierwszego i trzeciego zestawu danych. Ma on jednak problem z prawidłową klasyfikacją czwartego zestawu danych. Drugi zestaw klasyfikatorów dobrze odwzorowuje drugi zestaw danych. Nie radzi sobie on jednak z zestawem pierwszym i czwartym.

Zjawisko szumów:

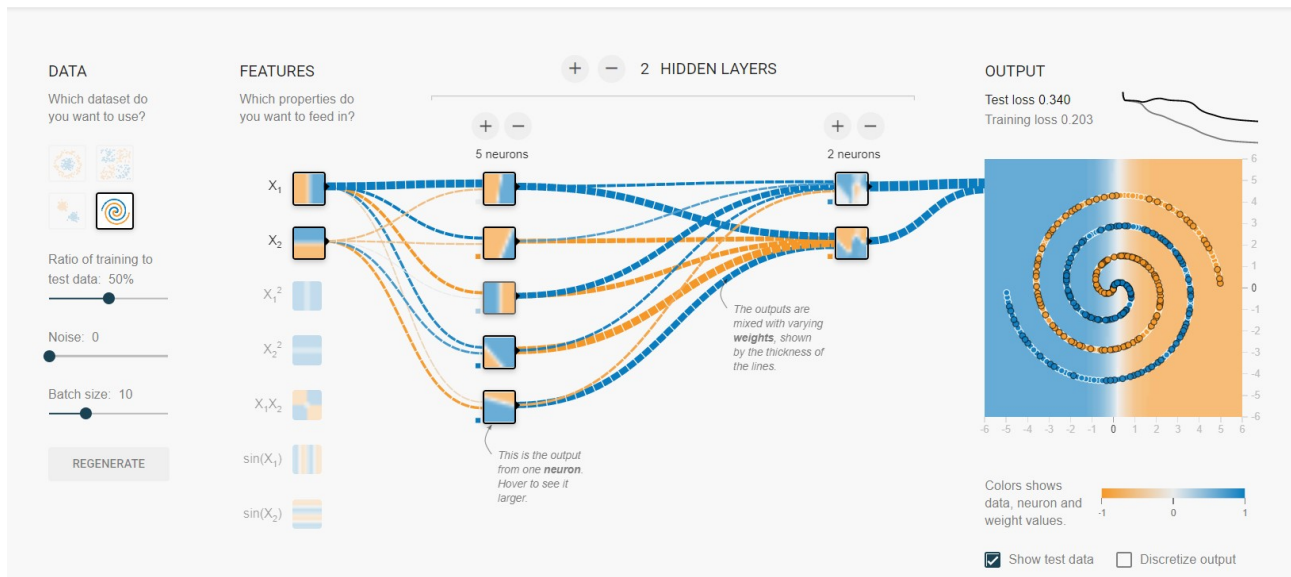
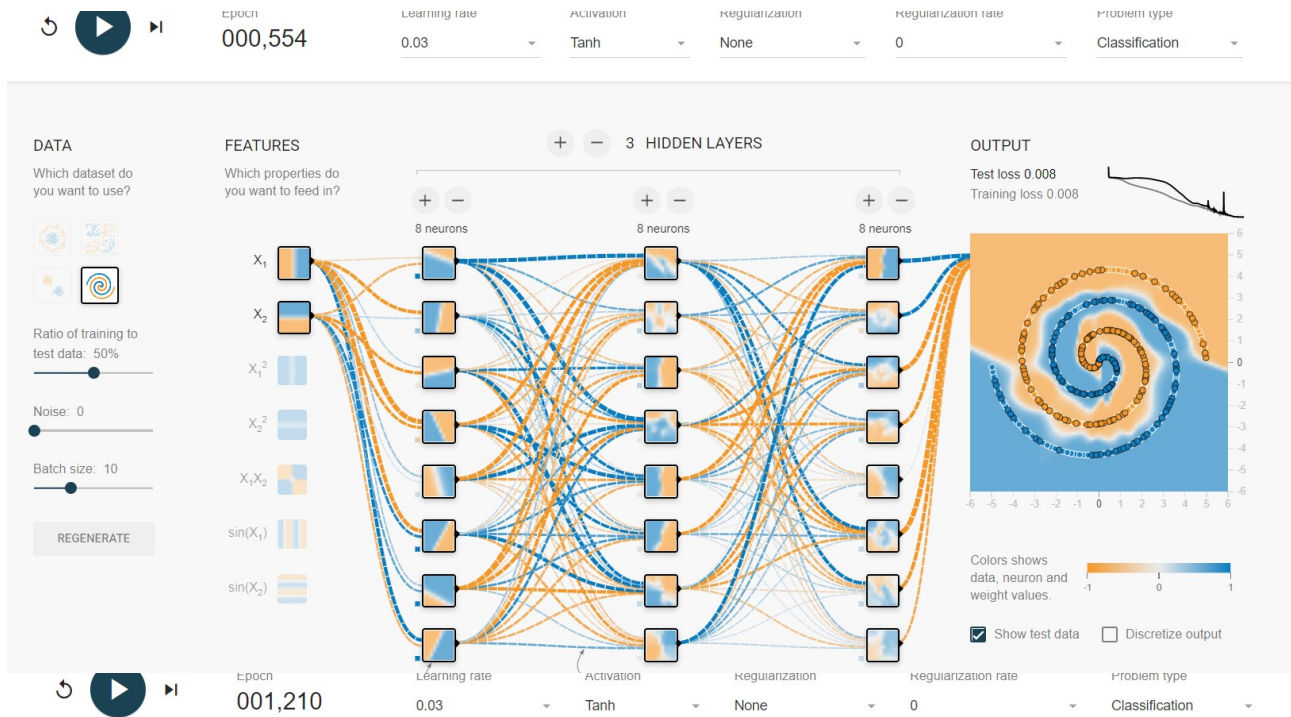




Wyraźnie widać, że zjawisko szumów pogarsza efekt klasyfikacji.  
Zastosowanie zbyt wielkiej liczby klasyfikatorów powoduje pogorszenie się klasyfikacji i duże fluktuacje niedopasowania klasyfikacji.



Zaszło tutaj zjawisko over fittingu, nasza sieć neuronowa wypracowała zależności nieprawdziwe dla drugiego zestawu danych.



Jak widać dla większej liczby warstw i neuronów klasyfikacja zachodzi znacznie szybciej i dokładniej. Zbyt mała ilość neuronów nie pozwala na poprawną klasyfikację.