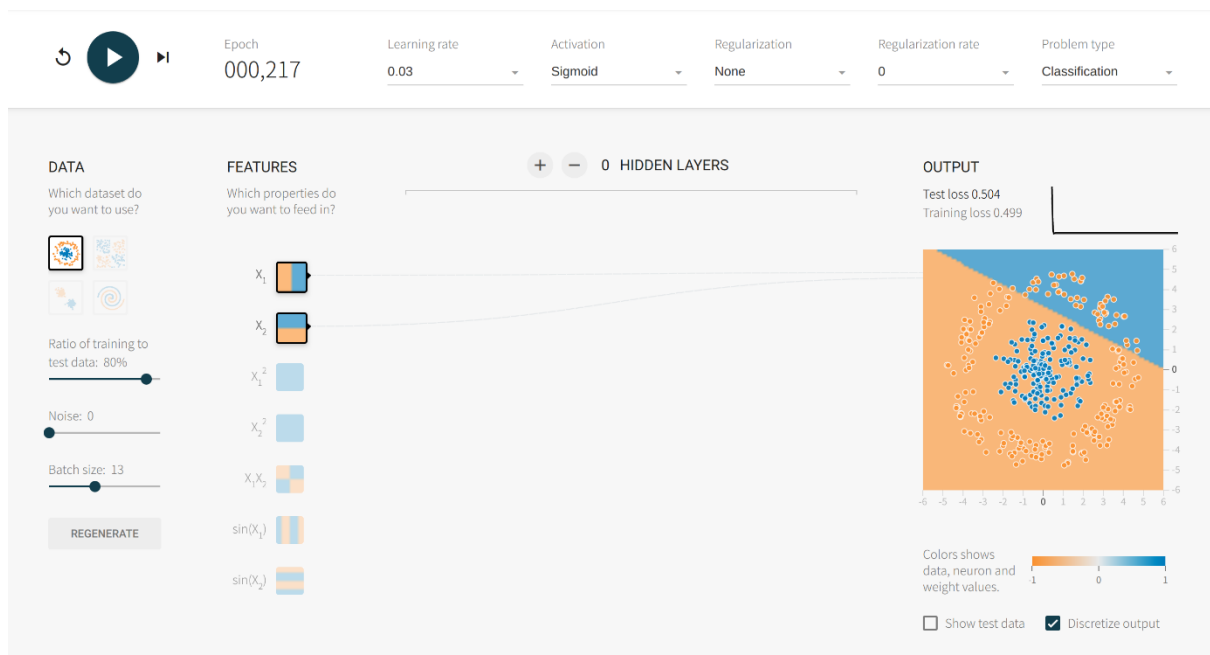
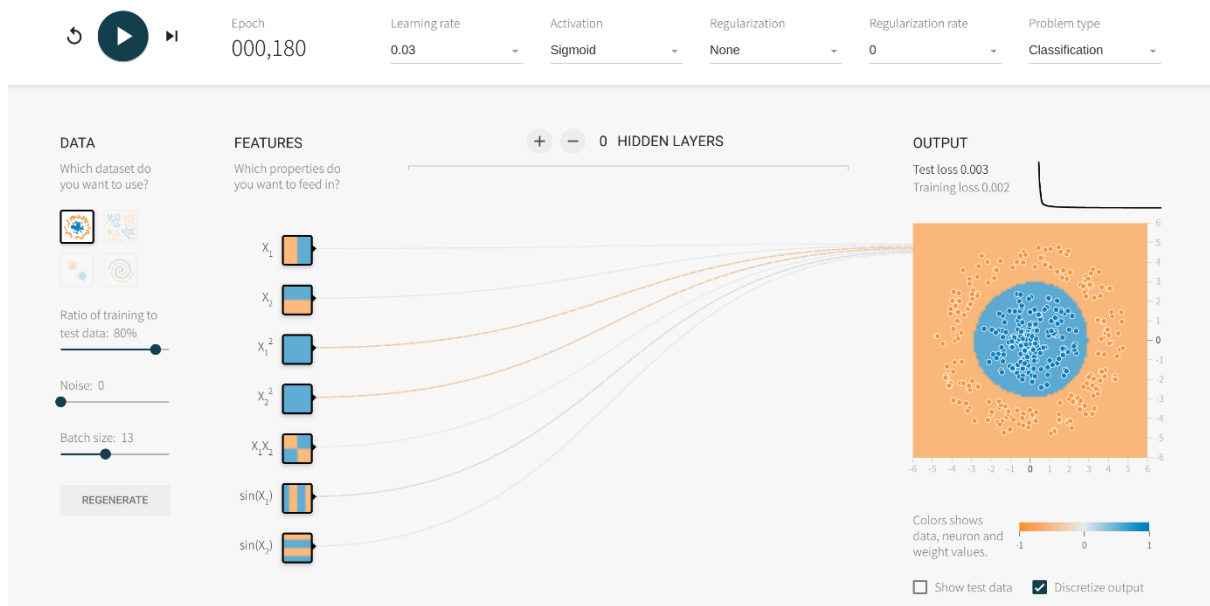


Aufgabe 1.1



Wenn die Features sehr einfach ausgewählt werden, kann die Klassifikation (ohne hidden Layer) nicht gut funktionieren, weil das Model theoretisch ein lineares Model ist. Im Vergleich dazu ist das Target (circular dataset) hoch nichtlinear.

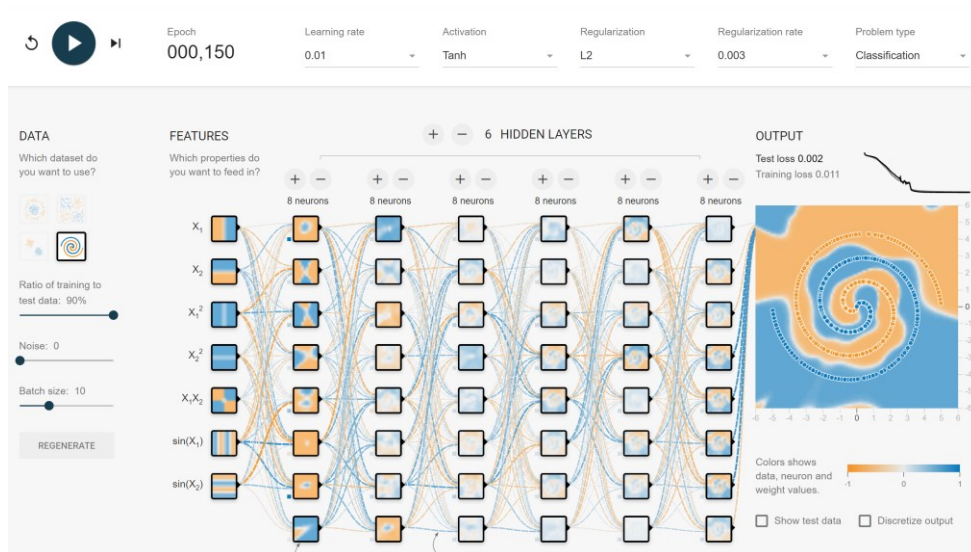


Wenn wir die Features mit hoher Ordnung und Kombination hinzugefügt haben, kann die Klassifikation gut mit Target anpassen.

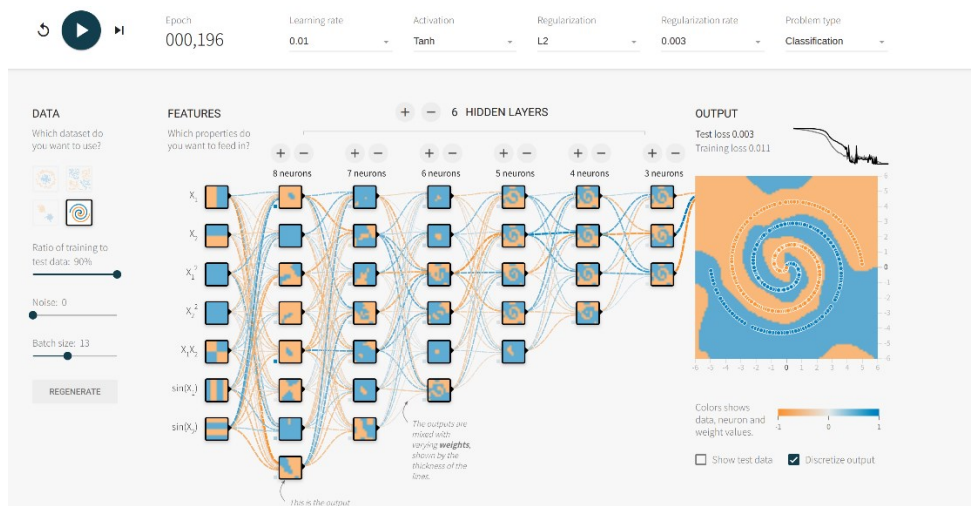
Grund liegt daran, dass die Features mit hoher Ordnung und Kombination die Nichtlinearität von Target erfüllen. Obwohl die Preceptron Architektur sehr einfach, kann es auch gut funktionieren.

Aufgabe 1.2

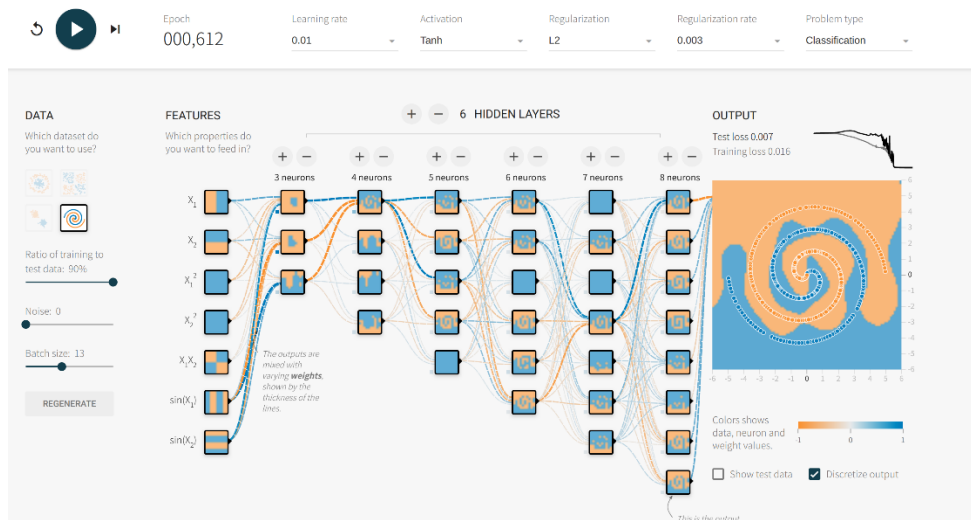
1. Same amount of neurons in every hidden layer



2. More neurons towards the input, less neurons towards the output.



3. Less neurons towards the input, more neurons towards the output.



Which scenario produces the best results on the test set?

Nach Loss Funktion hat erstes Model am besten Ergebnis. (Same amount of neurons in every hidden layer)

Which scenario converges the fastest?

Nach Epoche Zeit funktioniert das erste Model am besten. (Same amount of neurons in every hidden layer)

Explain in up to three sentences

Explain:

Das erste Model ist am komplexes in alle drei Model, deswegen kann es bessere Nichtlinearität haben.

Aufgabe 2.2

Configuration:

```
# Training
Perceptron.SGD(X=X_train, y=Y_train, alfa=0.003, size=10, epoch=800)
# after Training
```

Del set:

```
-----
accuracy before : 0.505941
Loss before : 782.960721
-----
accuracy after : 0.711695
Loss after : 458.172903
-----
```

Test

set:

```
-----
accuracy before : 0.512821
Loss before : 775.397387
-----
accuracy after : 0.676673
Loss after : 505.334551
-----
```