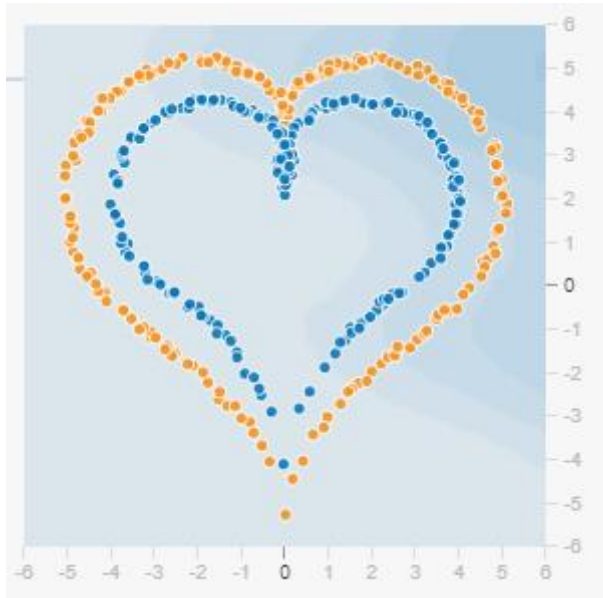
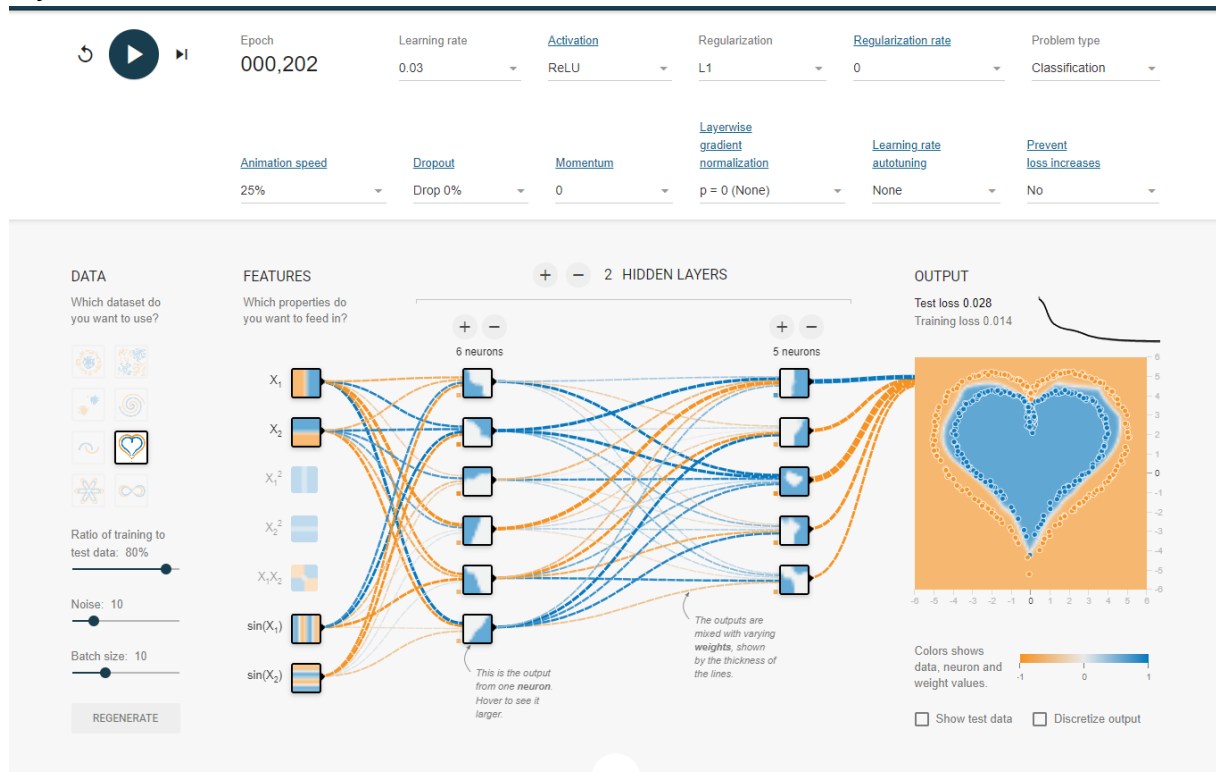


Deeper Playground

Wybrany element nr 1



Wykonana sieć:

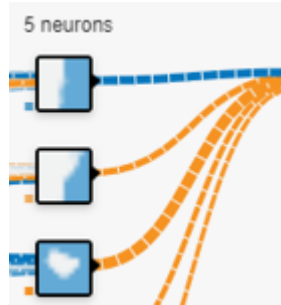


Odpowiedzi na pytania do elementu nr 1:

| |
|--|
| Pytanie: |
| <i>1.1 Jaką funkcję aktywacji wybrano i dlaczego? Czy można było zastosować inne, szybsze funkcje?</i> |
| <p>W powyższym rozwiązaniu została wybrana funkcja aktywacji ReLU, głównie ze względu na swoją prostotę obliczeniową i efektywność.</p> <p>Jako alternatywę można było zastosować funkcję Tanh, która działa dobrze dla nieliniowych danych, aczkolwiek nie dałaby lepszych wyników.</p> |
| <i>1.2 Jaką szybkość trenowania wybrano i dlaczego?</i> |
| <p>Dla rozwiązania wybrano <i>learning rate</i> 0.3 w celu uzyskania zrównoważonego tempa nauki. Zbyt mała wartość mogłaby spowolnić trening, a zbyt duża prowadzić do niestabilnej funkcji straty.</p> |
| <i>1.3 Ile parametrów w sumie sieć musi trenować (wszystkie wagi oraz bias).</i> |
| <p>Wejścia -> pierwsza warstwa ukryta (6 neuronów) liczba wag: $4 \times 6 = 24$ liczba biasów: 6 suma: 30 parametrów</p> <p>Pierwsza warstwa -> druga warstwa ukryta (5 neuronów) liczba wag: $6 \times 5 = 30$ liczba biasów: 5 suma: 35 parametrów</p> <p>Druga warstwa -> wyjście liczba wag: 5 liczba biasów: 1 suma: 6</p> <p>Całkowita liczba parametrów: 71</p> |
| <i>1.4 Ile połączeń jest w sieci?</i> |
| <p>W pierwszej warstwie: $4 \times 6 = 24$ połączenia W drugiej warstwie: $6 \times 5 = 30$ połączeń Wyjście: 5 Razem: 59 połączeń</p> |

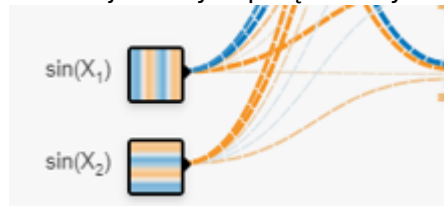
1.5 Które połączenia są najsilniejsze? Uzasadnić ilustracją wskazującą te połączenia.

Wyraźnie silniejsze połączenia widać w neuronach drugiej warstwy



1.6 Które połączenia są najslabsze? Uzasadnić ilustracją, wskazującą te połączenia.

Kilka najslabszych połączeń wychodzą z funkcji $\sin(X_1)$ i są ledwie widoczne



1.7 Czy można zastosować mniejszą ilość warstw, a jeśli nie to dlaczego.

W danym przypadku mniejsza ilość warstw nie jest zalecana, ponieważ dane są nieliniowe, a złożoność wymagała przynajmniej dwóch warstw do odpowiedniego dostosowania.

1.8 Czy można zastosować dropout? Jeżeli nie, to dlaczego nie?

Naszym celem było m.in. uzyskanie niskiej straty testowej i treningowej, a tak jak dropout służy do przeciwdziałania przeuczeniu, to nie było potrzeby w jego zastosowaniu

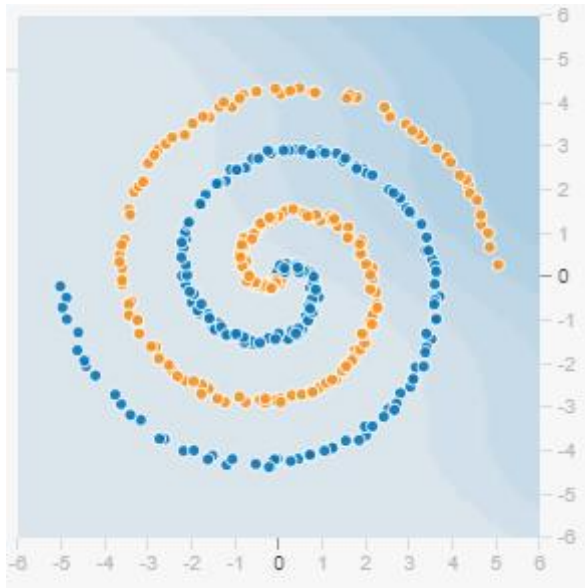
1.9 Czy konieczna była regularyzacja? Jeżeli nie to, dlaczego

W rozwiązaniu była zastosowana regularyzacja L1, ale z wartością 0. W naszym przypadku nie była ona konieczna, ponieważ różnica między training loss a test loss była mała i nie widać było oznak przeuczenia.

1.10 Czy można zmniejszyć ilość neuronów w warstwach? Jeśli nie, to dlaczego.

W praktyce zmniejszenie liczby neuronów skutkowało wzrostem błędu i niedopasowaniem oraz zwiększało wskaźniki training loss i test loss

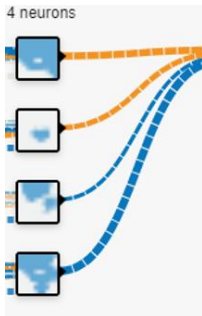
Wybrany element nr 2




Wykonana sieć:



Odpowiedzi na pytania do elementu nr 2:

| |
|---|
| Pytanie: |
| <i>1.1 Jaką funkcję aktywacji wybrano i dlaczego? Czy można było zastosować inne, szybsze funkcje?</i> |
| Dla zbudowania tej sieci również wykorzystano funkcję aktywacji ReLU ze względu na jej dobre efekty obliczeniowe. ReLU uznawana jest za najszybszą funkcję. Jako alternatywę można było poeksperymentować z Leaky ReLU. |
| <i>1.2 Jaką szybkość trenowania wybrano i dlaczego?</i> |
| Szybkość uczenia również została na poziomie 0.3. Ta umiarkowana wartość zapewniła stabilną i stosunkowo szybszą zbieżność podczas trenowania. |
| <i>1.3 Ile parametrów w sumie sieć musi trenować (wszystkie wagi oraz bias).</i> |
| <p>Wejścia -> pierwsza warstwa ukryta (8 neuronów) liczba wag: $6 \times 8 = 48$ liczba biasów: 8 suma: 56 parametrów</p> <p>Pierwsza warstwa -> druga warstwa ukryta (4 neurony) liczba wag: $8 \times 4 = 32$ liczba biasów: 4 suma: 36 parametrów</p> <p>Druga warstwa -> wyjście $(4 \times 1) + 1 = 5$ (4 wagi dla 1 neuronu plus 1 bias)</p> <p>Całkowita liczba parametrów: 97</p> |
| <i>1.4 Ile połączeń jest w sieci?</i> |
| <p>W pierwszej warstwie: $6 \times 8 = 48$ połączeń W drugiej warstwie: $8 \times 4 = 32$ połączeń Wyjście: 4 Razem: 84 połączenia</p> |
| <i>1.5 Które połączenia są najsilniejsze? Uzasadnić ilustracją wskazującą te połączenia.</i> |
| <p>Najsilniejsze połączenia widać na grubych liniach przy wyjściu drugiej warstwy.</p>  |

| |
|--|
| |
| <p><i>1.6 Które połączenia są najbliższe? Uzasadnić ilustracją, wskazującą te połączenia.</i></p> |
| <p>Najbliższe połączenia pokazują się na wyjściu z funkcji X_2^2</p>  |
| <p><i>1.7 Czy można zastosować mniejszą ilość warstw, a jeśli nie to dlaczego.</i></p> |
| <p>Najlepiej jeśli jest wystarczająca liczba warstw ukrytych, aby sieć mogła skutecznie rozwiązywać problem, chociażby ze względu na złożoność figury spirali. Być może można zastosować mniejszą liczbę warstw, ale przy użyciu większej liczby neuronów niż daje nam środowisko deppierplayground.</p> |
| <p><i>1.8 Czy można zastosować dropout? Jeżeli nie, to dlaczego nie?</i></p> |
| <p>Można zastosować dropout, aby uniknąć przeuczenia. Jednak w przypadku takiej niewielkiej sieci, dropout nie daje dobrych efektów uczenia. Dropout ma lepsze zastosowanie w przypadku bardziej rozbudowanych sieci. W tym przypadku dropout powoduje wzrost wartości funkcji straty.</p> |
| <p><i>1.9 Czy konieczna była regularyzacja? Jeżeli nie to, dlaczego</i></p> |
| <p>W danym rozwiązaniu użyto regularyzacji L1 (współczynnik 0), aby zmniejszyć wpływ mniej istotnych połączeń. Ale w tym przykładzie nie jest konieczne włączenie regularyzacji, wyniki są równie optymalne.</p> |
| <p><i>1.10 Czy można zmniejszyć ilość neuronów w warstwach? Jeśli nie, to dlaczego.</i></p> |
| <p>Teoretycznie jest to możliwe, pod warunkiem odpowiedniego dopasowania reszty parametrów. W praktyce przedstawione rozwiązanie było najbardziej optymalne.</p> |