

Projekt Zaliczeniowy Sieci Neuronowe 2022

Proszę o wykonanie poniższych zadań w języku programowania Python do 24.04 i wysłanie ich skompresowanych w repozytorium na moodle. Projekt wykonujemy w parach (jeżeli liczba osób w grupie jest nieparzysta dopuszczalny jest jeden zespół 3-osobowy lub 1 zespół 1-osobowy). Na przedostatnich zajęciach (28.04.2022) odbędzie się obrona projektów podczas których prowadzący ocenią projekt jednocześnie zadając do niego pytania. Poprawa projektów odbędzie się tydzień później. Za 1 termin mogą otrzymać Państwo maksymalnie 40 punktów (pozostałe 60 punktów zdobywają Państwo na teście). Oddawanie pracy w drugim terminie wiąże się z obniżeniem uzyskanej liczby punktów o 8 - jednakże nie wpływa to na próg zaliczenia wynoszący 20 punktów. Nieprzesłanie pracy w wyznaczonym terminie oznacza brak możliwości obrony projektu 28.04. Zabronione jest używanie innych bibliotek niż numpy i matplotlib. W przypadku gdy student chce użyć jakiejś innej biblioteki musi mieć na to zgodę prowadzącego. Użycie niezatwierdzonych bibliotek skutkuje oceną 0 punktów.

Perceptron

Zaimplementuj algorytm perceptronu w trybie PA i BUPA, który dla wektorów wejściowych $X^1 = [1,0,0]$, $X^2 = [1,0,1]$, $X^3 = [1,1,0]$, $X^4 = [1,1,1]$ znajduje granicę decyzyjną w postaci prostej w przestrzeni \mathbb{R}^2 . Przetestuj działanie algorytmu na przykładzie, gdy przynależność danego punktu do klasy (oznaczanej jako 1 lub 0) jest realizacją funkcji $d = x_1 \text{AND} x_2$, oraz $d = x_1 \text{XOR} x_2$. Stała uczenia równa jest 1. Niech każda iteracja działania obu algorytmów rysuje zmieniającą się granicę decyzyjną wykorzystując bibliotekę *matplotlib.pyplot*. Na wykresie zaznacz również położenie wektorów wejściowych. Zadbaj o dobry styl programowania i zabezpieczenie programu przed błędami, które mogą powstać przy implementacji. W przypadku funkcji XOR zaproponuj rozwiązanie problemu poprzez podniesienie wymiaru wektorów wejściowych przy pomocy radialnej funkcji bazowej RBF, a następnie użycie na takich danych algorytmu BUPA. Sprawdź, czy algorytm jest w stanie znaleźć granicę decyzyjną.

Sieć Hopfielda

Zaimplementuj algorytm odświeżania sieci Hopfielda w trybie synchronicznym. Program na początku swojego działania ma sprawdzić, czy spełnione są warunki stabilizacji sieci. Znajdź punkty stałe, sprawdź do jakich punktów stałych zbiegają pozostałe wektory wejściowe lub też wskaż, że dane wektory nie stabilizują się na danym stanie, a posiadają x okresową konfigurację. Przetestuj działanie sieci na danych z zadania 1 i 2 zaprezentowanych na laboratoriach.

Algorytm Propagacji Wstecznej

Zaimplementuj algorytm propagacji wstecznej dla funkcji XOR, omawiany na wykładzie dla dwóch neuronów w warstwie ukrytej. Funkcja aktywacji $f(t) = \frac{1}{1 + e^{-t}}$ ($\alpha = 1$) oraz stała uczenia $\eta = 0.5$ z wagami początkowymi:

$\omega_{2(0,1)} = 0.86,$	$\omega_{2(0,2)} = 0.82,$	$\omega_{3(0,1)} = 0.04,$
$\omega_{2(1,1)} = -0.16,$	$\omega_{2(1,2)} = -0.51,$	$\omega_{3(1,1)} = -0.43,$
$\omega_{2(2,1)} = 0.28,$	$\omega_{2(2,2)} = -0.89,$	$\omega_{3(2,1)} = 0.48,$

dla wektorów wejściowych: $X^1 = [1,0,0]$, $X^2 = [1,0,1]$, $X^3 = [1,1,0]$, $X^4 = [1,1,1]$ (kolejność wektorów przy energii cząstkowej) dla energii cząstkowej oraz całkowitej. Narysuj wykres jak zmienia się wartość energii sieci w trakcie jej nauczania w obu trybach.