Michalik Piotr

Wójcik Łukasz

Ewolucyjna konstrukcja modeli neuronowych

Zadanie:

Zbadać efektywność zastosowania programowania genetycznego do konstrukcji neuronowych sieci MLP (perceptronów wielowarstwowych) z pojedynczą warstwą ukrytą. Eksperymenty oprzeć na wybranym zestawie zadań regresji lub klasyfikacji.

# Specyfikacja projektu – definicja i zastosowanie

Projekt ten ma na celu przetestowanie możliwość stosowania algorytmów genetycznych przy modelowaniu sieci neuronowych MLP.

Algorytm genetyczny – jest to algorytm w którym istnieje pewna [populacja](https://pl.wikipedia.org/wiki/Populacja_(biologia)) osobników. Każdy z osobników ma przypisany pewien zbiór [informacji](https://pl.wikipedia.org/wiki/Informacja) stanowiących jego [genotyp](https://pl.wikipedia.org/wiki/Genotyp) - zbiór cech podlegających ocenie funkcji przystosowania (dopasowania). Funkcja dopasowania pełni rolę środowiska, oceniając potencjalne rozwiązania problemu. W zarysie ogólnym algorytm przyjmuje postać:

1. Generacja pierwotnej populacji osobników (losowych).
2. Ocena przystosowania każdego osobnika z populacji .
3. Powtarzanie poniższych elementów ewolucyjnych, aż do wystąpienia warunku stopu (dobry wynik lub osiągnięta maksymalna liczba powtórzeń - iteracje):
   * Wybrać grupę najlepiej przystosowanych osobników do reprodukcji.
   * Tworzenie nowego pokolenia osobników przez krzyżowanie i mutacje.
   * Ocena przystosowania nowych osobników.
   * Zastąpienie najgorzej dostosowanych osobników nową populacją.

Dodatkowo zdefiniowane zostają operatory:

Krzyżowanie – osobniki łączone są losowo w pary. Dla każdej pary ustalane jest (w drodze losowania, prawdopodobieństwo rzędu 20-50%), czy dojdzie do ich skrzyżowania. Jeśli tak, wylosowane zostaje miejsce (bit) w chromosomie jednego z rodziców, po czym fragmenty chromosomów są zamieniane miejscami poczynając od wylosowanego miejsca.

Mutacja – następuje wylosowanie osobnika, a następnie jeden z jego bitów. Wartość tego bitu zostaje zamieniona na przeciwną. Mutacja dotyka średnio 0.1% bitów w populacji.

Sieć MLP - wielowarstwowa sieć perceprtonowa, potocznie nazywana siecią neuronową. Jest to sieć co najmniej dwuwarstwowa zawierająca co najmniej dwa neurony McCullona-Pittsa. Jest to model implementujący algorytm [uczenia nadzorowanego](https://pl.wikipedia.org/wiki/Uczenie_nadzorowane). Sieć o odpowiedniej budowie może być wykorzystywana do klasyfikowania zbiorów które nie są liniowo rozróżnialne.

Na opis takiej sieci składa się wiele parametrów takich jak: liczba neuronów wejściowych, liczba neuronów wyjściowych, liczba warstw ukrytych (warstw pomiędzy wejściem i wyjściem), liczba neuronów w każdej warstwie ukrytej. Poza samą „fizyczną” charakterystyką zmianie podlegają parametry takie jak funkcja aktywująca, posiadanie biasu, czy algorytm uczenia sieci.

Problem -Dobór odpowiedniej architektury sieci jest kluczowy dla rozwiązania danego problemu klasyfikacji. Niestety nie ma jednoznacznej metody na ustalenie jaka charakterystyka sieci odpowiada za jaki aspekt problemu. Do tej pory w konstruowaniu sieci neuronowych posługiwano się własnym doświadczeniem lub metodą prób i błędów. Naszym zadaniem jest ustalenie czy istnieje możliwość wyewoluowania odpowiedniej sieci MLP za pomocą algorytmu genetycznego, tak aby dawała jak najlepsze wyniki dla danego problemu.

Postaramy się w genotypie osobnika w algorytmie genetycznym zawrzeć parametry sieci neuronowej. Następnie tak zdefiniujemy funkcje przystosowania, alby oceniała ona skuteczność sieci neuronowej na przykładowym zbiorze danych. Następnie ocenimy na podstawie testów jak skutecznie algorytm genetyczny jest w stanie wyewoluować sieć neuronową.

Elementy podlegające ewolucji – W naszym projekcie postanowimy przetestować zastosowanie algorytmu genetycznego przy ewolucji następujących parametrów sieci MLP:

1. Współczynnika nauki
2. Pędu (ang. Momentum)
3. Obciążenia (ang. Bias)
4. Liczby epok
5. Neuronów w warstwie ukrytej

Przed każdorazowym uruchomieniem algorytmu można będzie zdefiniować minimalne oraz maksymalne wartości każdego z parametrów, tak aby zoptymalizować czas obliczeń oraz mieć kontrolne nad podstawowym kierunkiem ewolucji. Np. Współczynnik nauki powinien należeć do przedziału [0.5 ; 0.00000001].

Reprezentacja osobnika – Zgodnie z założeniami algorytmu genetycznego, będziemy reprezentować pojedynczych osobników (instancje sieci MLP) binarnie. Każdy osobnik będzie opisany przez 37 bitów.

Bity [ 0 - 10 ] – reprezentują liczbę epok. 10 cyfrowa liczba binarna pozwala nam opisać wartości z przedziału [0 – 1023]. Zatem sieć może być uczona od 0 do 1023 epok. Przez epokę rozumiemy pełny cykl nauki sieci na wszystkich elementach ze zbioru uczącego.

Bity [ 11 – 20 ] – reprezentują współczynnik nauki. Również tutaj umożliwia nam to użycie wartości [0 – 1023], jednak my będziemy tak walidować te bity aby maksymalna liczba wynosiła 999. Dzięki temu, przedział [0 – 999] będzie odpowiadał współczynnikowi nauki z przedziału [0.0009 – 0.999].

Bit [21] – reprezentuje obciążenie sieci. Liczba 0 odpowiada wyłączonemu obciążeniu, a 1 włączonemu obciążeniu.

Bity [ 22 – 27 ] – reprezentują pęd uczenia sieci. 6 bitowa reprezentacja binarna daje nam wartości z przedziału [0 – 63]. Dzięki temu możemy ewoluować sieci ze współczynnikiem pędu z przedziału [0.01 – 0.63]. Mniejszy niż w przypadku współczynnika nauki przedział jest spowodowany tym, że zalecany w teorii uczenia MLP współczynnik pędu nie powinien przekraczać wartości 0.5.

Bity [ 28 – 37 ] – reprezentują liczbę neuronów w warstwie ukrytej. Również tutaj 10 cyfrowa reprezentacja daje nam możliwość sprawdzenia sieci o neuronach z przedziału [0 – 1023].

Zatem minimalna sieć będzie wyglądać następująco:

0000000000000000000000000000000000000

I będzie oznaczać sieć która się nie uczy (po wylosowaniu wag, nie następują żadne iteracje uczące). Sieć ma współczynnik nauki równy: 0.0009, współczynnik pędu: 0.01, wyłączone obciążanie oraz brak warstwy ukrytej (0 neuronów w tej warstwie).

Maksymalna sieć jednak będzie wyglądać:

1111111111111110011111111111111111111

I oznacza sieć która uczy się 1023 epoki, przy współczynniku nauki 0.999, współczynniku pędu: 0.63, oraz włączonym pędzie i 1023 neuronach w warstwie ukrytej.

Operatory genetyczne – Podczas ewolucji będziemy używać operatorów mutacji i krzyżowania. Będą się zachowywać tak jak opisane wcześniej operatory.

Eksperymenty – Podczas naszych eksperymentów będziemy bazować na przykładowych zadaniach klasyfikacji. Np. podział punktów na klasy. Wyniki będziemy oceniać na podstawie skuteczności klasyfikacji wyewoluowanych sieci.

Funkcja przystosowania – Funkcją przystosowania w naszym projekcie będzie funkcja przyjmująca jako argument reprezentacje sieci, a zwracająca procentowy wynik zawierający błąd sieci. Funkcja będzie zawierała w sobie algorytm który na podstawie reprezentacji sieci, obliczy wyniki tej sieci dla zbioru testowego i porówna je z wynikami idealnymi. Z tego następnie będzie obliczany błąd. Zatem funkcja przystosowania będzie funkcją którą nasz algorytm genetyczny będzie miał za zadanie minimalizować.