

SIECI NEURONOWE

Sprawozdanie - Ćwiczenie 2

Aleksander Poławski
Grupa - Poniedziałek 18:15
Prowadzący - mgr inż. Jan Jakubik

15 listopad, 2020

1 Cel ćwiczenia

Celem ćwiczenia drugiego laboratoriów kursu Sieci Neuronowe było zapoznanie się z siecią wielowarstwową, uczeniem sieci za pomocą algorytmu propagacji wstecznej w wersji klasycznej (minimalizacja błędu średniokwadratowego) oraz wpływem parametrów odgrywających istotną rolę w uczeniu sieci z propagacją wsteczną.

2 Plan ćwiczenia oraz badań

- a) implementacja sieci neuronowej MLP do rozpoznawania cyfr ze zbioru MNIST
- b) przeprowadzenie eksperymentów badających skuteczność uczenia zaimplementowanej sieci w zależności od zmiany następujących ustawień:
 - liczby neuronów w warstwie ukrytej
 - wartości współczynnika uczenia
 - wielkości paczki (batch)
 - zastosowanej funkcji aktywacji (tanh/ReLU)
 - wartości odchylenia standardowego przy inicjalizacji wag początkowych z rozkładu normalnego
- b) przeprowadzenie dodatkowych eksperymentów badających skuteczność uczenia zaimplementowanej sieci w zależności od zmiany następujących ustawień:
 - liczby warstw ukrytych

3 Opis zaimplementowanego programu

Program zaimplementowano w środowisku PyCharm w języku Python, korzystając między innymi z bibliotek Numpy do przetwarzania obliczeń macierzowych.

W jego skład wchodzi następujące elementy:

- klasa Loader - umożliwia wczytywanie, przechowywanie i konwersję zbiorów uczących i testowych. Zawiera proste funkcje pomagające stwierdzić poprawność wczytania zbiorów.
- klasa MLP - zawiera całą logikę tworzenia, ustawień, uczenia i testowania sieci MLP
- klasa MLP_Layer - zawiera całą logikę tworzenia, ustawień i działania poszczególnych warstw sieci
- plik main - manager programu - organizujący kolejność wykonywania zadań programu, zawierający predefiniowane testy potrzebne do wykonania badań przewidzianych w ćwiczeniu

Program w sposób prosty i intuicyjny umożliwia wykonanie wszystkich zaplanowanych w ćwiczeniu badań. Jego zaletą jest też możliwość definiowania dowolnej liczby warstw ukrytych (a nie tylko dwóch jak wymagano w treści zadania).

4 Badania

W poniższej sekcji zamieszczono i opisano wyniki badań.

4.1 Wpływ liczby neuronów w warstwie ukrytej na skuteczność uczenia MLP

a) Założenia

- liczba warstw ukrytych: [1]
- **zmienna liczba neuronów w warstwie ukrytej: [5], [7], [10], [25], [50]**
- funkcja aktywacji w warstwach ukrytych: [ReLU]
- inicjalizacja wag i biasów: [rozkład normalny]
- odchylenie standardowe w inicjalizacji wag: [0.1]
- wartość współczynnika uczenia α : [0.01]
- wielkość paczki (batch): [100]

b) Przebieg dla każdego ustawienia

- wykonanych zostało 10 procedur po 10 iteracji uczenia
- dokonano klasyfikacji obiektów zbioru testowego dla co drugiej iteracji
- obliczono średnią trafność klasyfikacji dla co drugiej iteracji ze wszystkich procedur uczenia
- wyliczono średni czas trwania iteracji

c) Otrzymane wyniki

liczba neuronów w warstwie ukrytej	średnia trafność klasyfikacji po iteracji				
	II	IV	VI	VIII	X
5	35.26%	52.04%	76.40%	79.18%	80.54%
7	72.68%	84.04%	86.46%	87.44%	88.21%
10	75.89%	85.34%	87.62%	88.53%	89.12%
25	82.85%	87.79%	89.25%	89.99%	90.49%
50	84.56%	88.45%	89.85%	90.72%	91.21%

Tabela 1: Wpływ liczby neuronów w warstwie ukrytej na skuteczność MLP

liczba neuronów w warstwie ukrytej	średni czas iteracji uczenia
5	13.84s
7	14.11s
10	15.88s
25	25.12s
50	34.09s

Tabela 2: Średni czas iteracji w zależności od liczby neuronów w warstwie ukrytej

d) Komentarz

- Badania ukazują, że większa liczba neuronów w warstwie ukrytej pozwala na szybsze osiągnięcie lepszego wyuczenia MLP.
- Warto jednak zauważyć, że im większa liczba neuronów tym dłużej trwa jedna iteracja uczenia (zwiększa się znacznie liczba operacji do wykonania), należy więc dobrać ustawienie w taki sposób, aby znaleźć balans pomiędzy skutecznością i obciążeniem komputera obliczeniami.
- Dla testowanych danych i ustawień jako najlepszy wybór wyznaczono liczbę 10 neuronów warstwy ukrytej - osiągnięto skuteczność porównywalną z ustawieniami o wyższej liczbie neuronów przy niskim czasie iteracji.

4.2 Wpływ współczynnika uczenia α na skuteczność uczenia MLP

a) Założenia

- liczba warstw ukrytych: [1]
- zmienna liczba neuronów w warstwie ukrytej: [10]
- funkcja aktywacji w warstwach ukrytych: [ReLU]
- inicjalizacja wag i biasów: [rozkład normalny]
- odchylenie standardowe w inicjalizacji wag: [0.1]
- **zmienna wartość współczynnika uczenia α : [0.01], [0.05], [0.10], [0.25], [0.50]**
- wielkość paczki (batch): [100]

b) Przebieg dla każdego ustawienia

- wykonanych zostało 10 procedur po 10 iteracji uczenia
- dokonano klasyfikacji obiektów zbioru testowego dla co drugiej iteracji
- obliczono średnią trafność klasyfikacji dla co drugiej iteracji ze wszystkich procedur uczenia

c) Otrzymane wyniki

współczynnik uczenia α	średnia trafność klasyfikacji po iteracji				
	II	IV	VI	VIII	X
0.01	75.89%	85.34%	87.62%	88.53%	89.12%
0.05	88.33%	89.96%	90.72%	90.88%	90.21%
0.10	89.85%	91.06%	91.41%	91.21%	91.35%
0.50	85.38%	86.01%	86.50%	86.53%	86.06%
0.70	86.30%	87.20%	86.99%	87.11%	87.01%

Tabela 3: Wpływ współczynnika uczenia α na skuteczność uczenia MLP

d) Komentarz

- Badania wskazują, że im wyższa wartość współczynnika uczenia tym szybciej sieć zostaje wyuczona (większe kroki w modyfikacji wag).
- Duży skok przy modyfikacji wag może jednak sprawić, że optymalna wartość wag będzie pomijana, a zatem sieć MLP nigdy nie osiągnie tak skutecznego wyuczenia jak przy mniejszym współczynniku α . Możemy to zjawisko zauważyć przy największym badanym współczynniku, gdzie co dwie iteracje otrzymujemy na zmianę lepszą i gorszą skuteczność.
- Ponownie należy więc uzyskać balans pomiędzy wydajnością procedury uczenia, a jej skutecznością.
- Inną możliwością jest wprowadzenie zautomatyzowanego, adaptowanego współczynnika uczenia np. poprzez implementację 'momentum' (ćwiczenie trzecie laboratorium).

4.3 Wpływ wartości parametru odchylenia standardowego w rozkładzie normalnym inicjalizacji wag na skuteczność uczenia MLP

a) Założenia

- liczba warstw ukrytych: [1]
- zmienna liczba neuronów w warstwie ukrytej: [10]
- funkcja aktywacji w warstwach ukrytych: [ReLU]
- inicjalizacja wag i biasów: [rozkład normalny]
- **zmienne odchylenie standardowe w inicjalizacji wag: [0.80], [0.40], [0.30], [0.10], [0.01]**
- wartość współczynnika uczenia α : [0.01]
- wielkość paczki (batch): [100]

b) Przebieg dla każdego ustawienia

- wykonanych zostało 10 procedur po 5 iteracji uczenia
- dokonano klasyfikacji obiektów zbioru testowego dla każdej iteracji
- obliczono średnią trafność klasyfikacji dla każdej iteracji ze wszystkich procedur uczenia

c) Otrzymane wyniki

odchylenie standardowe	średnia trafność klasyfikacji po iteracji				
	I	II	III	IV	V
0.80	17.33%	17.45%	17.53%	17.44%	18.01%
0.40	26.57%	27.56%	27.93%	28.10%	28.28%
0.30	40.65%	57.82%	67.42%	73.72%	77.45%
0.10	49.17%	71.58%	80.78%	84.88%	86.53%
0.01	21.27%	24.70%	60.69%	79.89%	83.96%

Tabela 4: Wpływ wartości parametru odchylenia standardowego w rozkładzie normalnym inicjalizacji wag na skuteczność uczenia MLP

d) Komentarz

- Im większe odchylenie standardowe tym większa rozbieżność i większa możliwa wartość inicjalizowanych losowo wag. Parametr wpływa więc silnie na początkową wydajność uczenia się sieci.
- Zbyt wysoka wartość ustawienia przy niskiej wartości współczynnika uczenia może sprawić, że sieć nie będzie w stanie wyuczyć się w pożądanym czasie z powodu zbyt mocno odbiegających od optymalnych wartości wylosowanych wag (i zbyt małego kroku uczenia, żeby je skorygować w kolejnych iteracjach).
- Ustawiając zbyt niską wartość tracimy natomiast szansę na wylosowanie chociaż częściowo dobrych wag, sprawiając, że wszystkie wartości są zbliżone do zera.
- Bezpieczniejszym wyborem wydaje się więc ustawienie zbyt niskiego odchylenia standardowego, niż zbyt wysokiego. Z wyników badań wynika, że przy zbyt wysokim współczynniku (0.8/0.4) sieć nie jest w stanie osiągnąć dobrej skuteczności do piątej iteracji, w przypadku zbyt niskiego współczynnika (0.01) sieć zaczyna od tak samo złej skuteczności, jednak do piątej iteracji uzyskuje pożądaną efekty.
- Istnieje także możliwość implementacji mechanizmów automatycznego określania optymalnych parametrów inicjalizowania wag (ćwiczenie trzecie laboratorium).

4.4 Wpływ funkcji aktywacji w warstwie ukrytej na skuteczność uczenia MLP

a) Założenia

- liczba warstw ukrytych: [1]
- zmienna liczba neuronów w warstwie ukrytej: [10]
- **zmienna funkcja aktywacji w warstwach ukrytych: [ReLU], [TanH]**
- inicjalizacja wag i biasów: [rozkład normalny]
- odchylenie standardowe w inicjalizacji wag: [0.1]
- wartość współczynnika uczenia α : [0.01]
- wielkość paczki (batch): [100]

b) Przebieg dla każdego ustawienia

- wykonanych zostało 10 procedur po 5 iteracji uczenia
- dokonano klasyfikacji obiektów zbioru testowego dla każdej iteracji
- obliczono średnią trafność klasyfikacji dla każdej iteracji ze wszystkich procedur uczenia

c) Otrzymane wyniki

funkcja aktywacji	średnia trafność klasyfikacji po iteracji				
	I	II	III	IV	V
ReLU	49.17%	71.58%	80.78%	84.88%	86.53%
TanH	45.54%	66.82%	74.81%	79.97%	82.44%

Tabela 5: Wpływ funkcji aktywacji w warstwie ukrytej na skuteczność uczenia MLP

d) Komentarz

- Wyniki wykazują, że użycie ReLU jako funkcji aktywacji jest w tym wypadku nieznacznie skuteczniejsze.
- W teorii, przewaga funkcji aktywacji ReLU rośnie w przypadku korzystania z większej liczby warstw ukrytych, ponieważ jest bardziej odporna na efekt zanikającego gradientu niż funkcja tangensa hiperbolicznego.

4.5 Wpływ wielkości paczki (batch) na skuteczność uczenia MLP

a) Założenia

- liczba warstw ukrytych: [1]
- zmienna liczba neuronów w warstwie ukrytej: [10]
- funkcja aktywacji w warstwach ukrytych: [ReLU]
- inicjalizacja wag i biasów: [rozkład normalny]
- odchylenie standardowe w inicjalizacji wag: [0.1]
- wartość współczynnika uczenia α : [0.01]
- **zmienna wielkość paczki (batch): [1], [25], [50], [75], [100]**

b) Przebieg dla każdego ustawienia

- wykonanych zostało 10 procedur po 10 iteracji uczenia
- dokonano klasyfikacji obiektów zbioru testowego dla co drugiej iteracji
- obliczono średnią trafność klasyfikacji dla co drugiej iteracji ze wszystkich procedur uczenia

c) Otrzymane wyniki

wielkość paczki (batch)	średnia trafność klasyfikacji po iteracji				
	II	IV	VI	VIII	X
1	87.09%	85.91%	84.66%	83.44%	82.46%
25	88.44%	90.22%	91.11%	91.69%	91.81%
50	85.42%	88.42%	89.69%	90.35%	90.77%
75	81.50%	87.08%	88.56%	88.52%	90.11%
100	75.28%	85.84%	88.02%	89.12%	89.72%

Tabela 6: Wpływ wielkości paczki (batch) na skuteczność uczenia MLP

d) Komentarz

- Im większa wielkość paczki, tym rzadziej modyfikowane są wagi, ale dokładniej (ponieważ modyfikacja bierze pod uwagę błąd z wielu wektorów wejściowych)
- Tym samym zwiększenie paczki spowoduje wydłużony, lecz dokładniejszy proces uczenia. Oznacza to wyuczenie po większej liczbie iteracji, ale z możliwością osiągnięcia bardziej optymalnych wag.
- W przypadku paczki o wielkości 1 przeuczenie (osiągnięcie optimum dla aktualnego ustawienia) występuje już po pierwszej iteracji (nieznacznie gorsze wyniki po każdej iteracji) i stosunkowo niska maksymalna osiągnięta skuteczność.
- Najlepszym wyborem w tym wypadku - patrząc na stosunek uzyskiwanych wyników do czasu uczenia - wydaje się ustawienie wielkości paczek po 25 wektorów. Chcąc jednak uzyskać lepsze wyniki w dłuższej perspektywie, mogłoby się okazać, że liczba paczek o wielkości od 50 do 100 byłaby jeszcze skuteczniejsza.

5 Badania dodatkowe

W poniższej sekcji zamieszczono i opisano wyniki badań dodatkowych (ponad program zadania).

5.1 Skuteczność uczenia MLP w zależności od różnej liczby warstw o zmiennej liczbie neuronów

a) Założenia

- **zmienna liczba warstw ukrytych:** [1], [2], [3], [4]
- **zmienna liczba neuronów w warstwie ukrytej:** [5], [7], [10]
- funkcja aktywacji w warstwach ukrytych: [ReLU]
- inicjalizacja wag i biasów: [rozkład normalny]
- odchylenie standardowe w inicjalizacji wag: [0.1]
- wartość współczynnika uczenia α : [0.05]
- wielkość paczki (batch): [25]

b) Przebieg dla każdego ustawienia

- wykonanych zostało 10 procedur uczenia, aż do momentu przeuczenia (do momentu kiedy błąd walidacyjny zaczął się zwiększać)
- wyznaczono średnią trafność klasyfikacji ze wszystkich procedur uczenia dla ostatniej (najskuteczniejszej) iteracji

c) Otrzymane wyniki

liczba warstw ukrytych	liczba neuronów w warstwach ukrytych	najlepsza skuteczność	iteracja wyuczenia
1	10	92.07%	4
2	10; 5	79.83%	2
2	10; 10	86.98%	2
3	10; 10; 10	82.02%	3
3	10; 10; 7; 5	11.04%	1

Tabela 7: Skuteczność uczenia MLP w zależności od różnej liczby warstw o zmiennej liczbie neuronów

d) Komentarz

- Niestety w wypadku tego ćwiczenia zwiększanie liczby warstw wynika szybszym przeuczeniem i gorszą maksymalną skutecznością.
- Być może wynika to z faktu, że badanie przeprowadzono z pozostałymi ustawieniami (współczynnik uczenia, odchylenia standardowe itd.) dopasowanymi w toku ćwiczenia tak, aby to właśnie jedna warstwa ukryta dawała najlepsze wyniki.
- Spersonalizowanie wszystkich pozostałych parametrów pod konkretną liczbę warstw mogłoby sprawić, że uzyskiwane wyniki byłyby wtedy lepsze.
- Potwierdzenie lub zaprzeczenie tej tezy będzie łatwiej uzyskać po ćwiczeniu trzecim, gdzie zaimplementowane zostaną zautomatyzowane metody doboru pewnych parametrów.

5.2 Badanie macierzy pomyłek dla wybranego ustawienia MLP

a) Założenia

- liczba warstw ukrytych: [1]
- liczba neuronów w warstwie ukrytej: [10]
- funkcja aktywacji w warstwach ukrytych: [ReLU]
- inicjalizacja wag i biasów: [rozkład normalny]
- odchylenie standardowe w inicjalizacji wag: [0.1]
- wartość współczynnika uczenia α : [0.05]
- wielkość paczki (batch): [25]

b) Przebieg dla każdego ustawienia

- wyuczono sieć MLP korzystając z powyższych ustawień
- utworzono macierz pomyłek dokonując klasyfikacji wektorów zbioru testowego

c) Otrzymana macierz pomyłek

oryginalne klasy	przewidywane klasy									
	0	1	2	3	4	5	6	7	8	9
0	939	0	4	2	0	5	7	4	19	0
1	0	1118	3	2	1	1	3	0	7	0
2	9	11	899	26	5	2	16	6	46	12
3	4	1	11	930	1	16	4	6	27	10
4	1	7	4	0	850	1	17	2	21	79
5	17	8	0	39	7	752	15	8	37	9
6	19	5	18	0	19	25	856	4	12	0
7	3	15	20	11	10	0	0	887	29	53
8	3	21	6	43	19	20	3	1	844	14
9	4	7	0	12	15	10	0	8	14	939

Tabela 8: Macierz pomyłek

d) Komentarz

- większość cyfr jest najczęściej mylona z cyfrą 8
- najczęściej poprawnie klasyfikowana jest cyfra 1
- najczęściej mylone cyfry to 4 z 9 i 8 z 3
- wynik badań potwierdza, że najczęściej mylone są cyfry o podobnym kształcie

6 Podsumowanie

Pomyślnie udało się zrealizować następujące wytyczne zadania:

- zaimplementowano elastyczną w konfiguracji i łatwą do rozbudowy w kolejnych zadaniach sieć MLP do rozpoznawania cyfr ze zbioru MNIST
- dodatkowo, zaimplementowano sieć w taki sposób, aby można było definiować dowolną liczbę warstw ukrytych
- wykonano wiele badań dyktowanych w treści zadania oraz opracowano ciekawe wyniki dokumentujące zdobytą w zadaniu wiedzę
- wykonano i opracowano dwa nieobowiązkowe (ale dotyczące zagadnień laboratorium) badania

W wyniku procesu wykonywania zadania rozwinęto swoją wiedzę na temat elementarnych pojęć dotyczących sieci neuronowych oraz nauczono się mechanizmów działania sieci MLP.