

Zadanie 5

Wielowarstwowy perceptron

Kacper Kania
23.04.2024

Treść zadania

Perceptrony wielowarstwowe (z ang. *Multilayer Perceptron* lub *MLP*) wszechobecne w dzisiejszym uczeniu maszynowym. Mimo że w powszechnych zadaniach klasyfikacji obrazów stosuje się głównie konwolucyjne sieci neuronowe, MLP dalej są głównym składnikiem Transformerów [1] (w modelach językowych) czy NeRFów [2] (do generowania obrazów). Poznać Państwo jak działa taki perceptron, jak wygląda jego uczenie oraz jak wpływają jego różne hiperparametry na jakość uczenia.

Państwa zadaniem jest:

- implementacja przejścia w przód,
- implementacja przejścia w tył (wsteczna propagacja lub *backpropagation*),
- implementacja stochastycznego gradientu schodzącego (SGD) do optymalizacji zdefiniowanej sieci neuronowej.

Do wykonania zadaniaa zalecam wykorzystanie operacji macierzowych w bibliotece `numpy` choć przetwarzanie sekwencyjne też powinno działać stosunkowo szybko.

Proszę zbadać następujące funkcje hiperparametry:

- liczba warstw ukrytych (w przypadku przetwarzania sekwencyjnego więcej niż dwie warstwy ukryte mogą znacząco spowolnić eksperymenty),
- wielkość paczki uczenia (*batch size*),
- dwie funkcje aktywacji dla warstw ukrytych: ReLU oraz sigmoid.

ReLU i jego gradient mają następujące postaci:

$$f(x) = \max(x, 0), \quad \frac{\partial f(x)}{\partial x} = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases}. \quad (1)$$

Jako końcową funkcję aktywacji, proszę wykorzystać funkcję sigmoid i funkcję straty MSE:

$$\text{MSE}(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2. \quad (2)$$

Typowo w takich zadaniach wykorzystuje się wieloklasową entropię krzyżową oraz funkcję softmax. Jednakże, softmax ma skomplikowany

gradient dla innych funkcji strat (skomplikowałoby to implementację), natomiast wieloklasowa entropia krzyżowa wymaga kilka sztuczek implementacyjnych dla uniknięcia błędów numerycznych.

Do pomiaru jakości uczenia proszę wykorzystać miary F_1 (z argumentem `average="micro"`) oraz `accuracy`.

Zbiór danych

Do eksperymentów proszę wykorzystać zbiór danych `digits` z `sklearn.datasets.load_digits`¹. Proszę założyć stały zbiór danych (np. podział 70% próbek treningowych oraz 30% próbek testowych).

Dodatkowe uwagi

Proszę pamiętać, że to zadanie jest realizowane w grupach **dwuosobowych**.

Bibliografia

- [1] A. Vaswani *et al.*, “Attention Is All You Need,” *Advances in neural information processing systems*, vol. 30, 2017.
- [2] P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, “NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis,” in *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2020.

¹https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_digits.html#sklearn.datasets.load_digits