

# Budowa aproksymatora do predykcji cen akcji

## Programowanie sieciowe

### Lista nr 3

rok akademicki 2018/2019, semestr letni

Kwiecień 2019 r.

- 
1. Skanujemy przestrzeń parametrów takich jak
    - algorytm  $A \in \{\text{adam}, \text{sgd}, \text{lbfgs}\}$ ,
    - liczba neuronów  $k \in \{10, 20, \dots, 300\}$ ,
    - parametr uczenia  $\eta = \{0.1, 0.01, 0.001, 0.0001\}$ .Skanowanie może zostać wykonane za pomocą polecenia `GridSearchCV` albo być zaimplementowane w 3 zagnieżdżonych pętlach.
  2. Dla każdego zestawu parametrów  $A, k, \eta$  trenujemy sieć za pomocą `MLPRegressor` i sprawdzamy jego działanie na zbiorze walidacyjnym (tzn. danych, które nie były użyte do treningu ani do wyliczania funkcji celu).  

```
for i:=1 to 100 do
```

    - (a) Dzielimy zbiór danych po wyłączeniu zbioru walidacyjnego część treningową i testową w stosunku 70% dane treningowe, 30% dane testowe za pomocą polecenia `sklearn.model_selection.train_test_split.html`
    - (b) Dla każdego podziału i dla aktualnego modelu zbieramy wartość funkcji celu oraz błąd na zbiorze walidacyjnym.

```
end for;
```
  3. Rozwiązaniem jest ten regressor, który ma najmniejszy średni błąd na zbiorze walidacyjnym.