

---

# Laboratorium

## Klasyfikacja zdjęć konwolucyjną siecią neuronową

---

Zanim zaczniesz przypomnij sobie wykład lub zajrzyj do samouczków z Internetu np.

- <https://indoml.com/2018/03/07/student-notes-convolutional-neural-networks-cnn-introduction/>
- <https://www.analyticsvidhya.com/blog/2018/12/guide-convolutional-neural-network-cnn/>

### Zadanie 1

Celem tego zadania jest prosta symulacja filtra (kernela, jądra) na wygenerowanym obrazku.

- a) Załaduj załączony plik **cnn.py** i uruchom go. Sprawdź jak zakodowany jest obrazek i co się na nim znajduje.
- b) Sprawdź jak będzie wyglądał obrazek po przetworzeniu przez jądro wykrywania pionowych krawędzi:

$$\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

z przesunięciem **s=1** (stride) i bez dodawania padding (wymiar obrazka zmniejszy się o 2 piksele).

Wyświetl ten obrazek.

- c) Obrazki po przetworzeniu jądrem są często aktywowane odpowiednią funkcją. Jak będzie wyglądał obrazek gdy dodatkowo przetworzymy go rektyfikatorem liniowym?

$$ReLU(x) = \begin{cases} 0 & \text{dla } x < 0 \\ x & \text{dla } x \geq 0 \end{cases}$$

- d) Wykonaj kroki b-d dla jądra wykrywającego krawędzie poziome:

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

- e) Wykonaj kroki b-d dla jądra wykrywającego krawędzie ukośne. Wykorzystaj operator Sobela (<https://pl.wikipedia.org/wiki/Sobel>)

## Zadanie 2

Celem zadania jest wykorzystanie sieci konwolucyjnych do rozpoznawania psów i kotów z obrazków. Wykorzystujemy do tego specjalistyczne paczki (np. keras, tensorflow, pytorch).

Paczka ze zdjęciami (miniaturki) została dołączona na stronie (*dogs-cats-mini.zip*) i zawiera dokładnie 12500 zdjęć kotów i 12500 zdjęć psów (razem około 26 MB)

Jaki jest plan działania:

- a) Zapoznaj się z wybranym samouczkiem np.
  - a. keras:
    - i. <https://machinelearningmastery.com/how-to-develop-a-convolutional-neural-network-to-classify-photos-of-dogs-and-cats/>
  - b. tensorflow:
    - i. <https://pythonprogramming.net/convolutional-neural-network-kats-vs-dogs-machine-learning-tutorial/>
    - ii. (można wyszukać inne)
- b) Załaduj bazę danych i dokonaj jej obróbki (przetworzenie obrazów, wyciągnięcie klasy cat/dog z nazwy obrazka).
- c) Podziel dane na zbiór testowy i treningowy, i być może też walidacyjny.
- d) Skonstruuj, wytrenuj model sieci konwolucyjnej na zbiorze treningowym. Sieć może mieć standardową, zaproponowaną w Internecie konfigurację.
- e) Podaj krzywą uczenia się dla zbioru treningowego i walidacyjnego.
- f) Podaj dokładność sieci na zbiorze testowym.
- g) Podaj macierz błędów na zbiorze testowym.  
(Pytania dodatkowe: Czy są koty przypominające psy, albo psy przypominające koty? Ile? Potrafisz może wskazać konkretne zdjęcia omyłkowo zakwalifikowanych zwierząt?)
- h) Powtórz parę razy eksperyment d-g z inną konfiguracją sieci (optimizer, funkcje aktywacji, inna struktura sieci, dropout, itp.). Wskaż jaka konfiguracja działała najlepiej i pokaż jej wyniki (krzywa uczenia się, dokładność, macierz błędów).