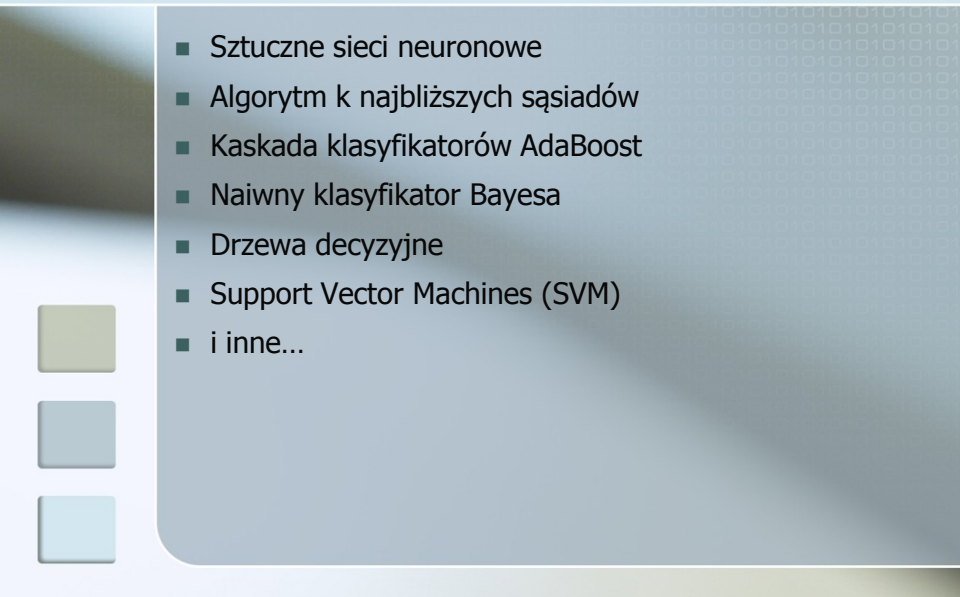


## Algorytmy decyzyjne będące alternatywą dla sieci neuronowych

Piotr Dalka

### Przykładowe algorytmy decyzyjne

- 
- Sztuczne sieci neuronowe
  - Algorytm k najbliższych sąsiadów
  - Kaskada klasyfikatorów AdaBoost
  - Naiwny klasyfikator Bayesa
  - Drzewa decyzyjne
  - Support Vector Machines (SVM)
  - i inne...

## Plan wykładu

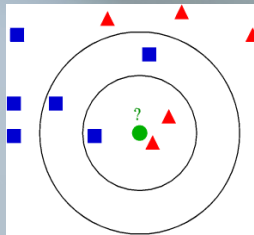
- Algorytm k najbliższych sąsiadów
- Kaskada klasyfikatorów AdaBoost
  - Działanie klasyfikatora i jego trening
  - Zastosowanie klasyfikatora do rozpoznawania obrazów

## Algorytm k najbliższych sąsiadów

- Jeden z najprostszych algorytmów decyzyjnych
- Zbiorem treningowym są wektory parametrów, stanowiące punkty w wielowymiarowej przestrzeni parametrów
- Badana próbka jest klasyfikowana jako należąca do tej klasy, która posiada w zbiorze treningowym największą ilość swoich „przedstawicieli” wśród k najbliższych sąsiadów próbki
- Miarą odległości jest zwykle metryka euklidesowa
- Parametr k jest ustalany z góry

## Algorytm k najbliższych sąsiadów

- Gdy  $k = 1$  (algorytm najbliższego sąsiada), badana próbka jest przypisywana do tej klasy, do której należy najbliższy niej wzorzec ze zbioru treningowego
- Im większe  $k$  tym algorytm jest bardziej odporny na szum, ale jednocześnie granice między klasami stają się bardziej rozmyte



## Cechy algorytmu

- Brak rzeczywistej fazy treningu
- Wysoka złożoność obliczeniowa procesu klasyfikacji
  - wymaga obliczenia odległości badanej próbki od wszystkich wzorców,
  - rośnie wraz ze wzrostem liczebności zbioru treningowego
- Wysoce wrażliwy na zaszumione lub nieistotne dane
- Klasa z większą ilością wzorców (w porównaniu z innymi klasami) ma tendencję do dominowania w wynikach klasyfikacji

## Plan wykładu

- Algorytm k najbliższych sąsiadów
- Kaskada klasyfikatorów AdaBoost
  - Działanie klasyfikatora i jego trening
  - Zastosowanie klasyfikatora do rozpoznawania obrazów

## Kaskada klasyfikatorów AdaBoost

- AdaBoost (Adaptive Boosting) jest heurystycznym, binarnym klasyfikatorem
- Jest klasyfikatorem silnym, który składa się z zestawu klasyfikatorów słabych, z których każdy kolejny jest optymalizowany do poprawnego rozpoznawania wzorców błędnie zakwalifikowanych przez wcześniejsze klasyfikatory
- Kaskada klasyfikatorów AdaBoost pozwala uzyskać większą skuteczność działania, niż pojedynczy silny klasyfikator
- Wymaga przygotowania zestawu treningowego z wzorcami pozytywnymi i negatywnymi

## Klasyfikator podstawowy (słaby)

- Podstawowy, słaby klasyfikator  $h_j$  wykorzystuje tylko jeden parametr  $f_j$  i zwraca wartość 1 lub 0 w zależności od tego, czy wartość parametru jest większa (mniejszy) od zadanego progu. Znak nierówności  $p_j$  oraz wartość progu  $\theta_j$  są ustalane w taki sposób, aby najdokładniej odseparować dwie rozpoznawane klasy

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$$

## Klasyfikator AdaBoost

- Klasyfikator składa się z wielu klasyfikatorów słabych
- W procesie treningu jako pierwszy klasyfikator słaby wybierany (dodawany) jest ten, który charakteryzuje się najmniejszym, ważonym błędem rozpoznawania wzorców treningowych
- Waga danego wzorca rośnie w przypadku jego błędnej detekcji przez klasyfikator słaby
- Każdy kolejno dodawany klasyfikator słaby jest zatem optymalizowany pod kątem rozpoznania wzorców, które sprawiły najwięcej problemów wcześniejszym klasyfikatorom

## Klasyfikator AdaBoost

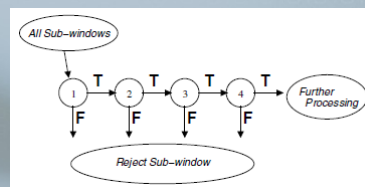
- Odpowiedź klasyfikatora AdaBoost dana jest wzorem:

$$h(x) = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0 & \text{otherwise} \end{cases}$$

gdzie  $\alpha_t$  jest proporcjonalne odwrotności błędowi detekcji ważonych danych wzorcowych przez klasyfikator słaby  $h_t$

## Kaskada klasyfikatorów AdaBoost

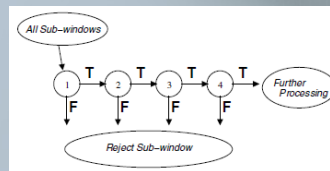
- Zestaw połączonych klasyfikatorów AdaBoost (członów)



- Pozytywny wynik klasyfikacji próbki wymaga pozytywnego rozpoznania kolejno przez wszystkie człony kaskady
- Nierozpoznanie próbki przez którykolwiek człon kaskady skutkuje negatywnym wynikiem klasyfikacji

## Tworzenie kaskady

- Dodawanie kolejnych parametrów (klasyfikatorów słabych) do danego członu kaskady trwa dopóki:
  - stopa błędów typu *false-positives* (wykrycie obiektu mimo jego faktycznego braku) nie spadnie poniżej założonego progu, np. 50%, lub
  - stopa błędów typu *false-negatives* (nie wykrycie obiektu mimo jego faktycznej obecności) nie wzrośnie powyżej założonego progu, np. 0,1%



## Tworzenie kaskady

- Dodawanie kolejnych członów kaskady trwa dopóki nie osiągnie się założonych stóp błędów *false-positives* i *false-negatives*
- Każdy kolejny człon kaskady jest trenowany w oparciu o wzorce pozytywne oraz te wzorce negatywne, które zostały „przepuszczone” przez wcześniejsze człony
- Taka strategia oznacza, że każdy kolejny człon kaskady zawiera coraz więcej parametrów (klasyfikatorów słabych), np. 1, 10, 25, 25, 50...
- Typowo stosuje się ok. 20-30 członów w kaskadzie, które zawierają łącznie jedynie kilka procent wszystkich możliwych parametrów (np. 6000 spośród 120 tys.)

## Cechy kaskady klasyfikatorów AdaBoost

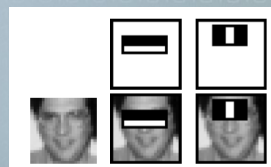
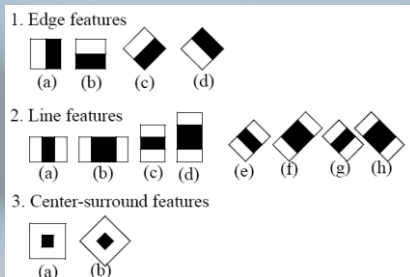
- Możliwość samodzielnego ustalenia kompromisu pomiędzy uzyskiwanymi wynikami a złożonością obliczeniową
- Bardzo wysoka wydajność algorytmu
- Automatyczny wybór optymalnych parametrów w procesie treningu klasyfikatora
- Zastosowanie klasyfikatorów AdaBoost połączonych w kaskadę pozwala jednocześnie uzyskać większą skuteczność klasyfikacji i większą wydajność
- Może być używana łącznie z innymi algorytmami decyzyjnymi w celu poprawy ich skuteczności
- Długi czas treningu

## Plan wykładu

- Algorytm k najbliższych sąsiadów
- Kaskada klasyfikatorów AdaBoost
  - Działanie klasyfikatora i jego trening
  - Zastosowanie klasyfikatora do rozpoznawania obrazów



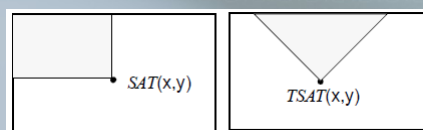
## Parametry wykorzystywane przy rozpoznawaniu obrazów



- Parametrami są różnice między sumą wartości pikseli (odcienie szarości) w białych i czarnych obszarach figur
- W obrazie 24x24 możliwe jest wyznaczenie 117941 parametrów

## Wyznaczanie parametrów obrazu

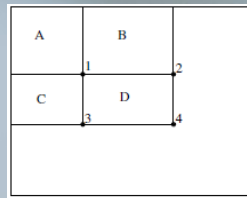
- Wartości każdego parametru są obliczane błyskawicznie i w identycznym czasie dla dowolnej skali i położenia na obrazie
- Wykorzystuje się do tego dwa tzw. obrazy zsumowane (ang. integral image) dla parametrów prostych (SAT) i obróconych o 45 stopni (TSAT):



- Pixel w obrazie zsumowanym (dla parametrów prostych) zawiera sumę wartości wszystkich pikseli znajdujących się powyżej i na lewo od niego.

## Wyznaczanie parametrów obrazu

- Wartości każdego parametru dla dowolnej skali i położenia można obliczyć błyskawicznie za pomocą kilku odwołań do tablicy w pamięci
  - np. suma wartości pikseli w dowolnym prostokącie wymaga odczytania 4 wartości z obrazu zsumowanego



- $D = 4 + 1 - (2 + 3)$

## Rozpoznawanie obiektów w obrazie

- Kaskada klasyfikatorów AdaBoost jest trenowana z wykorzystaniem zestawu kilku tysięcy wzorców pozytywnych i negatywnych o jednej, stałej i niewielkiej rozdzielczości, np. 24x24 piksele
- Jest w stanie rozpoznawać w obrazie obiekty o rozmiarach równych lub większych od rozmiaru wzorców

## Rozpoznawanie obiektów w obrazie

- W procesie detekcji okno, w którym prowadzona jest klasyfikacja, przesuwane jest po obrazie wejściowym z zadany krokiem, a następnie skalowane w górę o zadany współczynnik
- W czasie klasyfikacji zawartości każdego okna, skalowaniu nie ulega obraz wejściowy, lecz jedynie parametry klasyfikatora
- W praktyce w procesie detekcji zdecydowana większość okien jest odrzucana w pierwszym drugim członie kaskady, wobec czego dla każdego okna wymagane jest obliczenie średnio około 20-30 parametrów.

