

Klasyfikator złożony - idea

- ensemble/kombinacja klasyfikatorów
- Podstawowa idea:
 - podziel zbiór danych na k części
 - użyj ich do nauczania k klasyfikatorów
 - nowy rekord prześlij do zbudowanych klasyfikatorów
 - zintegruj odpowiedzi k klasyfikatorów (np. głosowanie większościowe)
- Różne metody konstrukcji zbiorów treningowych i agregacji odpowiedzi

Klasyfikator złożony - błąd

- Niższy błąd klasyfikacji gdy:
 - klasyfikatory bazowe są niezależne
 - klasyfikatory bazowe mają błąd mniejszy od klasyfikatora losowego

Klasyfikator złożony – podejścia

- Modyfikacja zbiorów treningowych:
 - bagging
 - boosting
- modyfikacja zbioru atrybutów zbiorów treningowych:
 - lasy losowe
- modyfikacja algorytmu klasyfikatora (dla tego samego zbioru danych treningowych)
- modyfikacja wartości etykiet klas decyzyjnych
 - error correcting output coding

Bagging

- agregacja bootstrapowa
- niech zbiór wejściowy ma n przykładów
- powtarzamy k razy:
 - próbujemy n razy zbiór wejściowy otrzymując zbiór treningowy (n przykładów, mogą się powtarzać)
 - w wylosowanym zbiorze jest średnio 63,2% elementów zbioru oryginalnego
 - elementy niewylosowane tworzą zbiór testowy
- uzyskane k zbiorów używamy do konstrukcji k klasyfikatorów głosujących większością
- metoda dobra dla klasyfikatorów wrażliwych na zaszumione dane
- zwiększa odporność na zjawisko przeuczenia (w przeciwieństwie do boostingu)

Boosting

- zbiory treningowe tworzone przez losowanie ze zwracaniem
- zmienia się prawdopodobieństwo wylosowania przykładu (początkowo równe dla wszystkich)
- Iteracyjny schemat działania:
 - losowanie ze zwracaniem (z uwzględnieniem wag)
 - uczenie klasyfikatora
 - ocena na oryginalnym zbiorze
 - aktualizacja wag przykładów (inkrementacja dla niepoprawnie sklasyfikowanych, dekrementacja dla poprawnie sklasyfikowanych)
- Nowe przypadki są klasyfikowane na podstawie ważonego (waga odpowiada funkcji trafności klasyfikatora) głosowania klasyfikatorów
- Wiele zaproponowanych algorytmów, np. AdaBoost