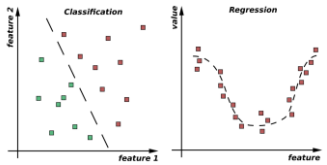
Przetwarzanie sygnałów i identyfikacja - moduł AI, Wykład 3

Klasyfikacja i regresja

Ziemowit Dworakowski
Akademia Górniczo-Hutnicza,
Katedra Robotyki i Mechatroniki

1

Klasyfikacja i regresja - ponownie



Klasyfikacja – przypisanie obiektu do określonej **klasy**

(Problem binarny - obiekt albo należy albo nie należy do klasy)

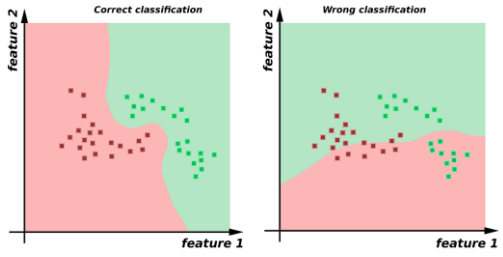
Regresja– przypisanie **wartości** do każdego obiektu

(Problem „ciągły" – wartość można przypisać z dowolną dokładnością)

- Obie są zawyczaj problemami wielowymiarowymi
- Obie bazują na **cechach** (ang. ***features***) – wysokiego lub niskiego poziomu
- Obie są bardzo podobne (praktycznie te same metody w obu przypadkach)

2

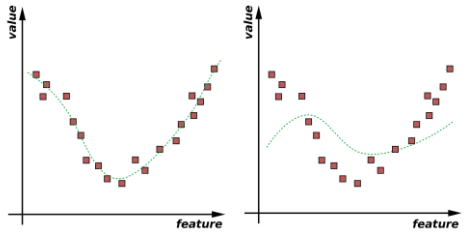
Klasyfikacja i regresja - ponownie



Uczenie klasyfikatora: Minimalizowanie **błędu klasyfikacji** na pewnym zbiorze **danych uczących**

3

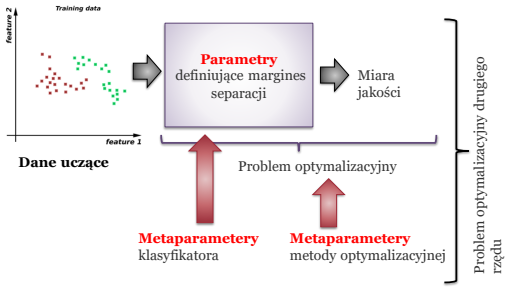
Klasyfikacja i regresja - ponownie



Uczenie regresora: Minimalizowanie **błędu regresji** na pewnym zbiorze **danych uczących**

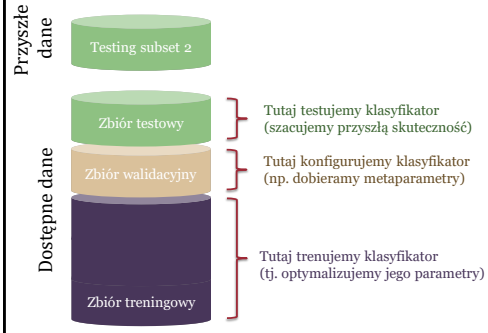
4

Problem treningu klasyfikatora



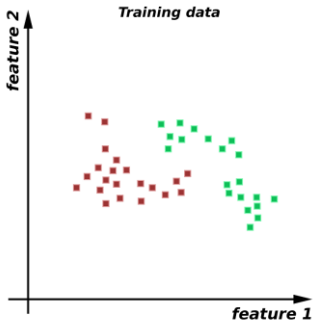
5

Podział danych



6

Klasyfikacja i regresja: Dane w klastrach



„Zgrupowania" danych nazywamy klastrami.

Dane zgrupowane w klastrach mają zalety

Mają też wady

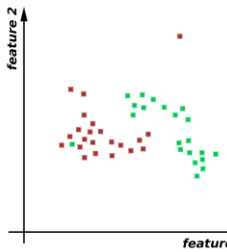
Od czego zależy, czy dane będą wykazywały tendencje do „grupowania"?

7

Klasyfikacja i regresja: Outliery

Pojedyncze obserwacje „odstające" od pozostałych nazywamy **outlierami**.

Obecność outlierów wynika zazwyczaj z:
- błędów pomiarowych
- błędów w etykietowaniu

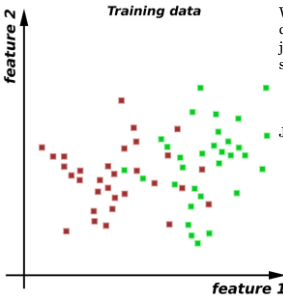


Najczęściej outlierów się pozbywamy na etapie wstępnego przetwarzania danych. Niektóre z nich nie są jednak łatwe do znalezienia.

Istnieją sposoby na radzenie sobie z nimi również na etapie projektowania algorytmów klasyfikacyjnych

8

Klasyfikacja i regresja: Nakładające się dane (*overlap*)



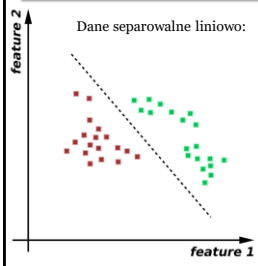
W praktycznych sytuacjach klasy bardzo często nakładają się na siebie – tzn. w jednym obszarze przestrzeni cech znajdują się dane z kilku różnych klas.

Jak można poradzić sobie z tym problemem?

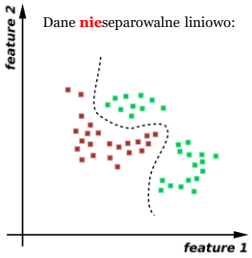
9

K i R: liniowa lub nieliniowa separowalność klas

Dane separowalne liniowo:



Dane **nie**separowalne liniowo:



10

K i R: Przeuczenie (*overfitting*)

Przeuczenie (overfitting) oznacza, że system klasyfikacji lub regresji osiąga bardzo dużą skuteczność na zestawie danych uczących oraz dużo niższą na zestawie testowym

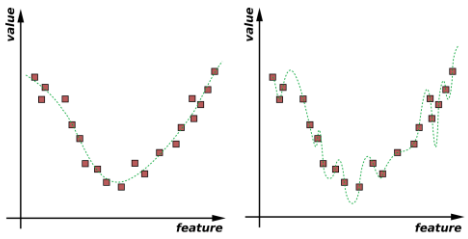
Innymi słowy:

System „zapamiętuje" dane uczące nie posiadając zdolności **uogólniania**

11

K i R: Przeuczenie (*overfitting*)

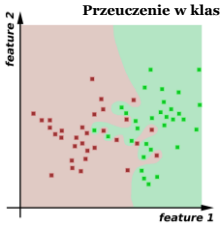
Przeuczenie w regresji:



12

K i R: Przeuczenie (*overfitting*)

Przeuczenie w klasyfikacji:



Jak unikać przeuczenia (Jak efektywnie **generalizować**) ?

- Podzbiór danych uczących wyłącznie do oceny skuteczności
- Odpowiednia ilość danych uczących
- Wczesne zatrzymanie uczenia, regularyzacja

13

K i R: „*Curse of dimensionality*”

Poziom trudności zadania klasyfikacji lub regresji rośnie wykładniczo wraz ze wzrostem ilości wymiarów przestrzeni cech

(Wymagane jest wykładniczo więcej danych aby „rozsądnie” wypełnić przestrzeń i umożliwić naukę i rozpoznawanie wzorców)

Jak sobie radzić z „klątwą wielowymiarowości”?

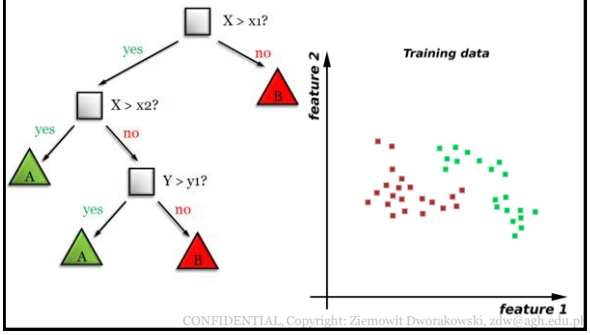
14

Podstawowe algorytmy

15

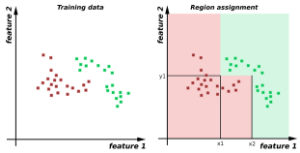
Drzewa decyzyjne

Klasyfikacja z wykorzystaniem DD:
X i Y to cechy, A i B to klasy



16

Drzewa decyzyjne - podsumowanie



+ Proste, intuicyjne, łatwe do zaimplementowania, szybkie
+ Łatwe do konfiguracji dla dwu lub trójwymiarowych problemów

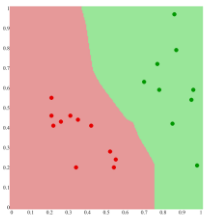
- Trudne do ręcznej konfiguracji w przypadku większej ilości wymiarów
- Automatyczne algorytmy nie są zbyt skuteczne
- Wrażliwe na przeuczenie i obecność outlierów

17

Metoda k-najbliższych sąsiadów
(k Nearest Neighbors, kNN)

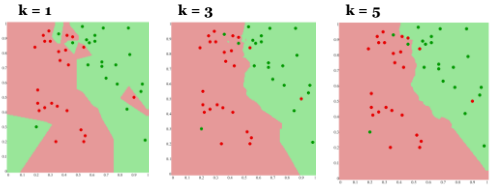
Aby ocenić przynależność nowej (niesklasyfikoweh) próbki:

1. **Znajdź jej k (nieparzyste) najbliższych sąsiadów**
2. **Wybierz najpopularniejszą etykietę**
3. **Przypisz nowej próbce wybraną etykietę**



18

Metoda *k-nearest neighbors* (kNN)



19

Metoda *k-nearest neighbors* (kNN), podsumowanie

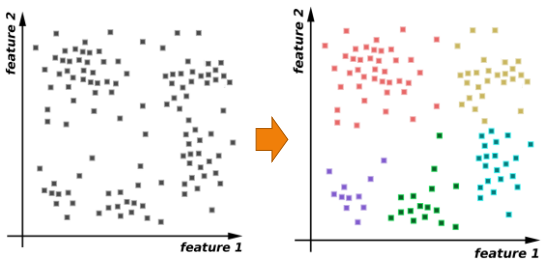
+ Intuicyjna
+ Łatwa do zaimplementowania
+ Dla *gęstych* danych odporna na outliery (jeśli k > 3)
+ Odporna na przeuczenie

- Bardzo wrażliwa na ilość wymiarów, źle działa w wielowymiarowych przestrzeniach
- Czasochłonna obliczeniowo, wymaga dużo pamięci

20

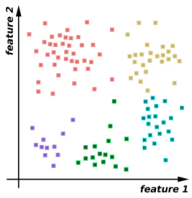
Klasteryzacja (*Clustering*) danych

Klasteryzacja *(Clustering)* polega na przypisaniu każdemu punktowi danych **etykiety klastra** (poza etykietą klasy) – używając algorytmu działającego *bez nauczyciela* (*ang.* ***unsupervised***)



21

Klasteryzacja (*Clustering*) danych

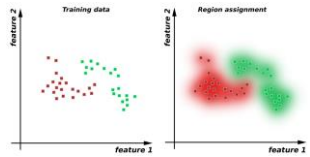


+ Oszczędza pamięć
+ Efektywna dla danych zgrupowanych w klastrach
+ Intuicyjna
+ Niewrażliwa na outliery
+ Niewrażliwa na przeuczenie (zazwyczaj)

- Trudna optymalizacja
- Nietrywialne określanie ilości klastrów
- Słabo działa, jeśli dane nie są wyraźnie pogrupowane
- Nie pozwala na skomplikowane marginesy separacji

22

Klasyfikatory statystyczne, podsumowanie

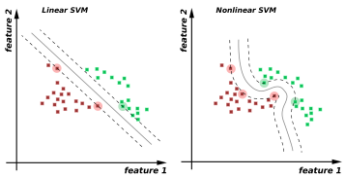


+ Intuicyjne
+ Niewrażliwe na outliery
+ Niewrażliwe na przeuczenie
+ Zapewniają miarę pewności klasyfikacji

- Wrażliwe na wysoką wymiarowość danych
- To grupa metod. Konkretne metody z grupy znacznie się różnią
- Złożone obliczeniowo i (czasem) pamięciowo (nieraz nawet bardziej niż kNN!)

23

Support Vector Machine (*SVM*)

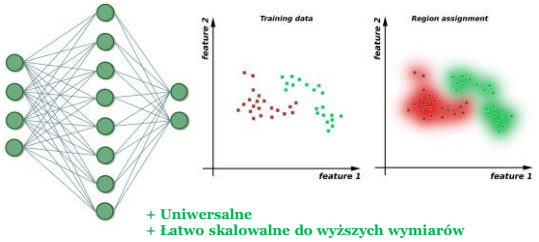


+ (Bardzo!) szybka nauka
+ Rozsądny wybór marginesu separacji
+ Podlega łatwemu skalowaniu w wyższą liczbę wymiarów

- Wymaga doświadczenia w konfiguracji
- Ignoruje gęstość danych (opiera się wyłącznie na granicach klastrów)

24

Sztuczne sieci neuronowe



+ Uniwersalne
+ Łatwo skalowalne do wyższych wymiarów
+ Dobra miara pewności klasyfikacji
+ Uwzględniają gęstość danych

- Wymagają doświadczenia w konfigurowaniu
- Wrażliwe na przeuczenie
- Niedeterministyczne

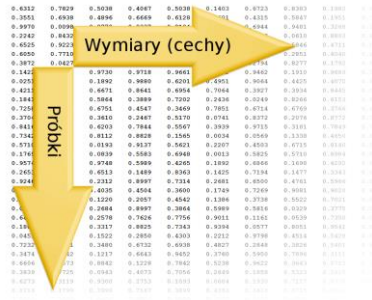
25

Classifiers and regressors comparison

Klasyfikator	Wymagana pamięć	Złożoność treningu	Prędkość „operacyjna”	Klasyfikacja i regresja?	Wrażliwość na ilość wymiarów	Trudność konfiguracji	Przeuczenie	Skuteczność
Drzewo dec.								
kNN								
Statystyczne								
Klasteryzacja								
SVM								
ANN								
...								
...								

26

Redukowanie ilości danych



27

Redukowanie ilości próbek

W ogólności *„Nie da się mieć ZA DUŻO danych”...*
(z teoretycznego punktu widzenia)

Jednakże, z uwagi na ograniczenia pamięciowe i skończoną moc obliczeniową czasami musimy okroić posiadany zbiór danych poprzez jego losowe próbkowanie, klasteryzację lub aproksymację rozkładu danych

28

Redukcja ilości wymiarów

*„**Curse of dimensionality**”* nakłada górne „miękkie ograniczenie” na ilość wymiarów: Więcej wymiarów wymaga większej ilości próbek.

Nawet jeśli mamy bardzo dużo danych, zazwyczaj lepiej jest ograniczyć ilość wymiarów tak bardzo, jak to możliwe
(więcej algorytmów do wyboru, większa skuteczność, krótsze obliczenia)

W ogólności, im mniej wymiarów tym lepiej. Staramy się wybrać najmniejszą ilość wymiarów która daje satysfakcjonujący rezultat.

29

Redukcja wymiarowości

Ogólne reguły:

- **Cechy powinny być znaczące**
(Każda z nich powinna wpływać pozytywnie na rezultat. Jeśli „odłączenie” którejś cechy nie zmienia znacząco wyniku, cecha jest niepotrzebna.

- **Cechy powinny być nieskorelowane (niezależne)**

- **Jak najmniejsza ilość cech**

30

Testowanie metod

- Standardowo, dane dostępne dzieli się na podzbiory losowo. Jeśli dane są zbierane w ramach osobnych programów eksperymentalnych warto zadbać o to, aby osobne podzbiory również były zbierane w formie osobnych programów eksperymentalnych.

- Podzbiory zawsze muszą być rozłączne (aby było możliwe wykrycie przeuczenia zarówno na etapie nauki jak i doboru metaparametrów)

- Jeśli chcemy „rozmnożyć" dane do nauki, robimy to zawsze **PO** podziale danych na podzbiory – aby ten sam punkt danych nie znalazł się kilkukrotnie w kilku podzbiorach

- Metody niedeterministyczne testujemy wielokrotnie

- Oceniamy średnią skuteczność, ale też skuteczność w każdej klasie z osobna

31

Przetwarzanie sygnałów i identyfikacja - moduł AI, Wykład 3

Klasyfikacja i regresja

1) **Podstawowe pojęcia: klastry danych, outliery, nakładanie się klas, liniowa i nieliniowa klasyfikacja, przeuczenie, klątwa wielowymiarowości**
2) **Jak redukować złożoność problemu?**
3) **Jakie są przykłady algorytmów klasyfikacyjnych? Jak działają?**
4) **Jak oceniać algorytmy klasyfikacyjne?**

32