Eksploracja danych
1. Wprowadzenie

Piotr Szwed
Katedra Informatyki Stosowanej AGH
2021

Informacje o przedmiocie

- 7 wykładów (opisy wybranych zagadnień i algorytmów)
- 14 laboratoriów
  - Część I (głównie tematy z poprzedniego przedmiotu)
    - Oprogramowanie Weka (użycie GUI: Explorer i KnowledgeFlow)
    - Java – programowe wywołanie funkcji biblioteki Weka [raczej niewiele]
    - Python + matplotlib + SciPy + ScikitLearn (w zależności od konfiguracji dostępnej w laboratorium). Gotowe fragmenty kodu.
  - Część II (nowe tematy)
    - Głównie Python
    - W trakcie przygotowania
- Zaliczenie
  - Na podstawie wykonanych ćwiczeń i kolokwium

Literatura

- Podręczniki
  1. Ian H. Witten, Eibe Frank, Mark A. Hall: Data Mining Practical Machine Learning Tools and Techniques Third Edition
  2. Tadeusz Morzy: Eksploracja danych, PWN 2013
  3. Daniel T. Larose, Discovering Knowledge In Data An Introduction To Data Mining
  4. Daniel T. Larose Data Mining Methods And Models/Metody i modele eksploracji danych
  5. Jiawei Han, Micheline Kamber, Data Mining: Concepts and Techniques Second Edition
  6. Oded Maimon, Lior Rokach: Data Mining and Knowledge Discovery Handbook
- Zasoby internetowe
  1. https://scikit-learn.org/stable/
  2. Tutorial Slides by Andrew Moore http://www.autonlab.org/tutorials
  3. Andrew Ng: Machine Learning, https://www.coursera.org/learn/machine-learning
  4. Emily Fox, Carlos Guestrin: Machine Learning Specialization, https://www.coursera.org/specializations/machine-learning

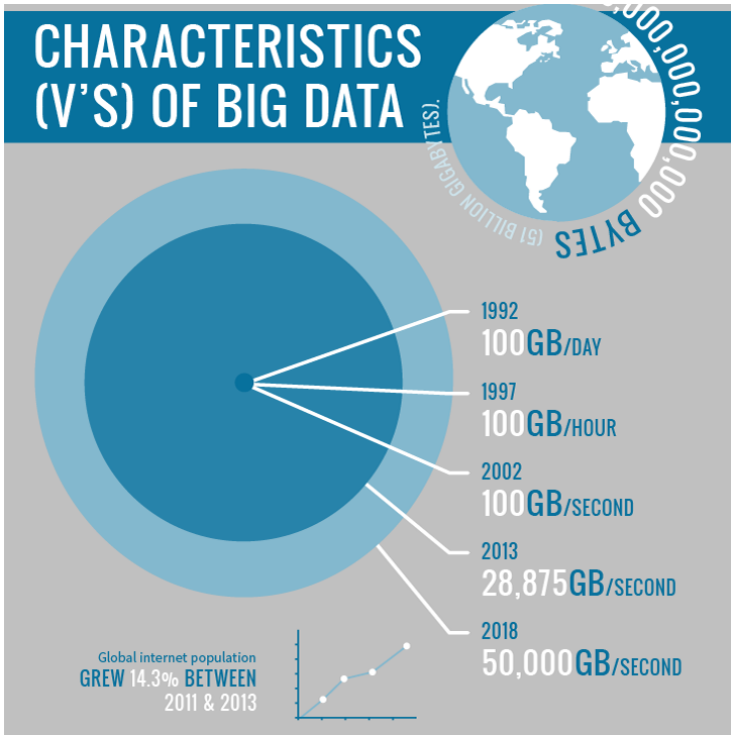
Wprowadzenie

Codziennie wytwarzane jest około $2.5\ 10^{18}$ bajtów danych

- dane operacyjne firm
- aktywność internetowa
- dane z urządzeń mobilnych
- pomiary z rozproszonych czujników

Oczekuje się, że odkrywanie i wykorzystanie użytecznych informacji zawartych w tych danych pozwoli na:

- poprawę efektywności działania przedsiębiorstw
- zmianę sposobu nawiązywania relacji między firmami i klientami (rekomendacja produktów i usług)
- poprawę jakości życia (opieki medycznej, organizacji transportu, itp.)
- zmniejszenie strat i ograniczenie marnotrawstwa energii, materiałów



[http://www.vcloudnews.com/every-day-big-data-statistics-2-5-quintillion-bytes-of-data-created-daily/]

Wybrane firmy wykorzystujące eksplorację danych

Na podstawie obecności na kaggle

Powiązane pojęcia

Uczenie maszynowe (machine learning, ML) obejmuje analizę, projektowanie i rozwijanie algorytmów, które rozszerzają możliwości komputerów o uczenie bez konieczności jawnego ich zaprogramowania [Arthur Samuel].

Metody statystyczne zajmują się modelami probabilistycznymi; wiele metod DM/ML to przeniesienie metod znanych ze statystyki

Eksploracja danych (data mining, DM) to proces, który mając na wejściu dane ma na celu wyodrębnienie z nich wiedzy lub interesujących wzorców. Podczas tego procesu używane są algorytmy uczenia maszynowego.

Metody statystyczne

data mining =
statistics + marketing

Ian H. Witten, Eibe Frank, Mark A. Hall Data Mining
Practical Machine Learning Tools and Techniques

Glossary

Machine learning	Statistics
network, graphs	model
weights	parameters
learning	fitting
generalization	test set performance
supervised learning	regression/classification
unsupervised learning	density estimation, clustering
large grant = $1,000,000	large grant= $50,000
nice place to have a meeting: Snowbird, Utah, French Alps	nice place to have a meeting: Las Vegas in August

[http://statweb.stanford.edu/~tibs/stat315a/glossary.pdf]

Podział metod eksploracji danych

- **Odkrywanie asocjacji** (*association mining*) – metody odkrywania zależności lub korelacji w zbiorach danych. Wynikiem są tzw. reguły asocjacyjne
- **Klasyfikacja i predykcja** (*classification and prediction*) – metody doboru modeli lub funkcji pozwalających na przewidywanie wartości wyjściowych (etykiet klas lub wartości ciągłych)
- **Grupowanie** (*clustering*) – podział zbioru obiektów na grupy mające podobne cechy.
  - maksymalizacja podobieństwa wewnątrz grupy
  - minimalizacja podobieństwa pomiędzy obiektami należącymi do różnych grup
- **Analiza sekwencji i przebiegów czasowych**
  - znajdowanie charakterystycznych podciągów
  - klasyfikacja i grupowanie
  - znajdowanie trendów, podobieństw, anomalii

Podział metod eksploracji danych

- **Odkrywanie charakterystyk** – znajdowanie zwięzłych opisów ogólnych własności klas obiektów, np. kombinacji wartości najważniejszych cech opisujących obiekt klasy. Wykorzystywane do różnicowania.
- **Eksploracja tekstu i danych półstrukturalnych** (*text mining*):
  - klasyfikacja treści
  - grupowanie podobnych dokumentów
  - analiza sentymentu
- **Rekomendacja** – metody rekomendacji mają na celu określenie obiektów, którymi użytkownik mógłby być zainteresowany
  - na podstawie danych historycznych dotyczących wybieranych wcześniej przedmiotów
  - podobieństw pomiędzy użytkownikami
- **Wykrywanie anomalii** – wykrywanie obiektów osobliwych, które odbiegają od ogólnego modelu danych lub modeli klas

Specyficzne zastosowania

- **Eksploracja sieci www**
  - indeksowanie stron
  - metody określania ich ważności
  - semantyczne kwerendy
  - lokalizacja reklam internetowych
- **Eksploracja grafów i sieci społecznościowych**
- **Eksploracja danych multimedialnych**:
  - indeksowanie obrazów i wideo
  - grupowanie i klasyfikacja
  - skróty wideo
- **Eksploracja danych przestrzennych**
  - grupowanie i klasyfikacja podobnych obiektów lub obszarów ze względu na różne cechy: gęstość zabudowy, roślinność, zanieczyszczenia, itp.

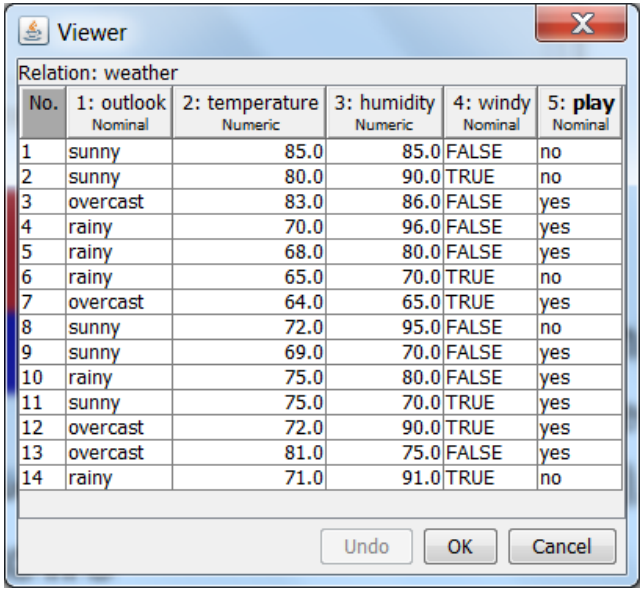
Typy danych

Dane dla algorytmów uczenia mają najczęściej postać tabeli:

- Wiersze zawierają instancje (obserwacje)
- Kolumnom są przypisane atrybuty określonego typu

Typy atrybutów:

- **Numeryczne**
  - dyskretne (liczby całkowite)
  - ciągłe (liczby rzeczywiste)
- **Kategoryczne** (symbole wyliczeniowe)
  - Porządkowe (ordinal),
    np.: low, medium, high
  - Symboliczne, nominalne (nominal),
    np.: sunny, overcast, rainy
- Dodatkowo
  - Daty (rodzaj danych numerycznych)
  - Dane tekstowe (raczej podlegające dalszemu przetwarzaniu, np. teksty są przekształcane do postaci wektora częstości wystąpień słów)



No.	1: outlook Nominal	2: temperature Numeric	3: humidity Numeric	4: windy Nominal	5: **play** Nominal
1	sunny	85.0	85.0	FALSE	no
2	sunny	80.0	90.0	TRUE	no
3	overcast	83.0	86.0	FALSE	yes
4	rainy	70.0	96.0	FALSE	yes
5	rainy	68.0	80.0	FALSE	yes
6	rainy	65.0	70.0	TRUE	no
7	overcast	64.0	65.0	TRUE	yes
8	sunny	72.0	95.0	FALSE	no
9	sunny	69.0	70.0	FALSE	yes
10	rainy	75.0	80.0	FALSE	yes
11	sunny	75.0	70.0	TRUE	yes
12	overcast	72.0	90.0	TRUE	yes
13	overcast	81.0	75.0	FALSE	yes
14	rainy	71.0	91.0	TRUE	no

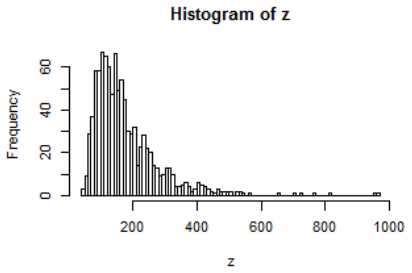
Proces odkrywania wiedzy

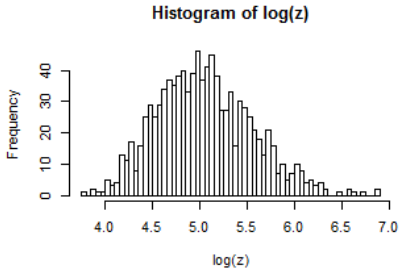
- **czyszczenie danych** (data cleaning) – usuwanie danych niepełnych, niepoprawnych
- **integracja danych** (data integration) – łączenie danych z heterogenicznych źródeł w jeden zbiór
- **selekcja danych** (data selection) – wybór danych istotnych z punktu widzenia analizy
- **konsolidacja i transformacja** (data consolidation & transformation) – przekształcenie do postaci wymaganej przez metody eksploracji danych (algorytmy uczenia)
- **eksploracja danych** (data mining) - odkrywanie wiedzy w postaci użytecznych wzorców lub modeli
- **ocena** (evaluation) – ocena jakości wzorców lub modeli
- **wizualizacja** (presentation) – prezentacja wyników

[T.Morzy: Eksploracja danych, PWN 2013]

Data Wrangling

1. Usuwanie zbędnych kolumn (atrybutów)
   1. ID, nieużywane nazwy
   2. Dużo brakujących wartości
   3. Mała wariancja
2. Usuń wiersze (obserwacje z pustymi wartościami)
3. Usuń szum
   1. Zastąp błędne wartości przybliżeniami
   2. Usuń dane odstające (ang. outliers)
   3. Znormalizuj rozkłady danych
4. Uzupełnij (ang. impute) brakujące wartości
5. Transformacja danych kategorycznych i dat
6. Transformacja cech: dyskretyzacja (ang: binning), skalowanie, normalizacja
7. Redukcja wymiarowości (usuwanie skorelowanych wartości, wybór cech, PCA)
8. Wydzielenie zbioru służącego do budowy modelu (uczącego) i jego walidacji (testowego)





http://theprofessionalspoint.blogspot.com/2019/03/data-wrangling-techniques-steps.html

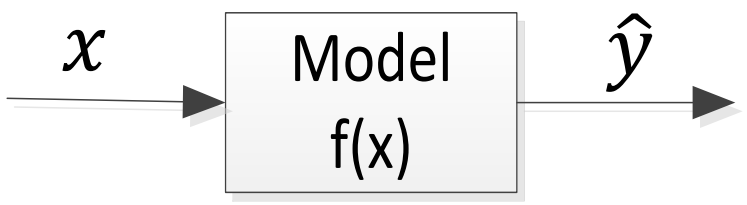
Rodzaje zagadnień 1

Uczenie nadzorowane, predykcja

- Zbiór uczący: $\mathrm{D} = \{(x_i, y_i)\}_{i=1,m}$
- (x_i, y_i) – obserwacje, instancje
- x_i - n-wymiarowy wektor wartości nazywanych cechami (ang. feature) lub atrybutami
- y_i - wartość wyjściowa, odpowiedzi, decyzyjna,

Klasyfikacja: $y_i \in \{1, \ldots, C\}$ – skończony zbiór wartości

Regresja: $y_i \in \boldsymbol{R}$



Wynikowy model można ocenić, np. dostarczając na wejście kolejne instancje x_i ze zbioru D i porównując: $\hat{y}$= $f(x_i)$ z y_i

Rodzaje zagadnień 2

sqft_living	sqft_lot	floors	price
1180	5650	1	221900
2570	7242	2	538000
770	10000	1	180000
1960	5000	1	604000
1680	8080	1	510000
5420	101930	1	1230000
1715	6819	2	257500
1060	9711	1	291850
1780	7470	1	229500

- Zmienna wyjściowa jest typu numerycznego.
- Cel: przewidywanie cen sprzedaży domów
- Zagadnienie: **regresja**

outlook	temp	humidity	windy	play
sunny	85	85	FALSE	no
sunny	80	90	TRUE	no
overcast	83	86	FALSE	yes
rainy	70	96	FALSE	yes
rainy	68	80	FALSE	yes
rainy	65	70	TRUE	no
overcast	64	65	TRUE	yes
sunny	72	95	FALSE	no
sunny	69	70	FALSE	yes
rainy	75	80	FALSE	yes
sunny	75	70	TRUE	yes
overcast	72	90	TRUE	yes
overcast	81	75	FALSE	yes
rainy	71	91	TRUE	no

- Zmienna wyjściowa jest typu kategorycznego
- Cel: dokonanie wyboru pomiędzy *yes* i *no* dla różnych kombinacji danych wejściowych
- Zagadnienie: **klasyfikacja**

Rodzaje zagadnień 3

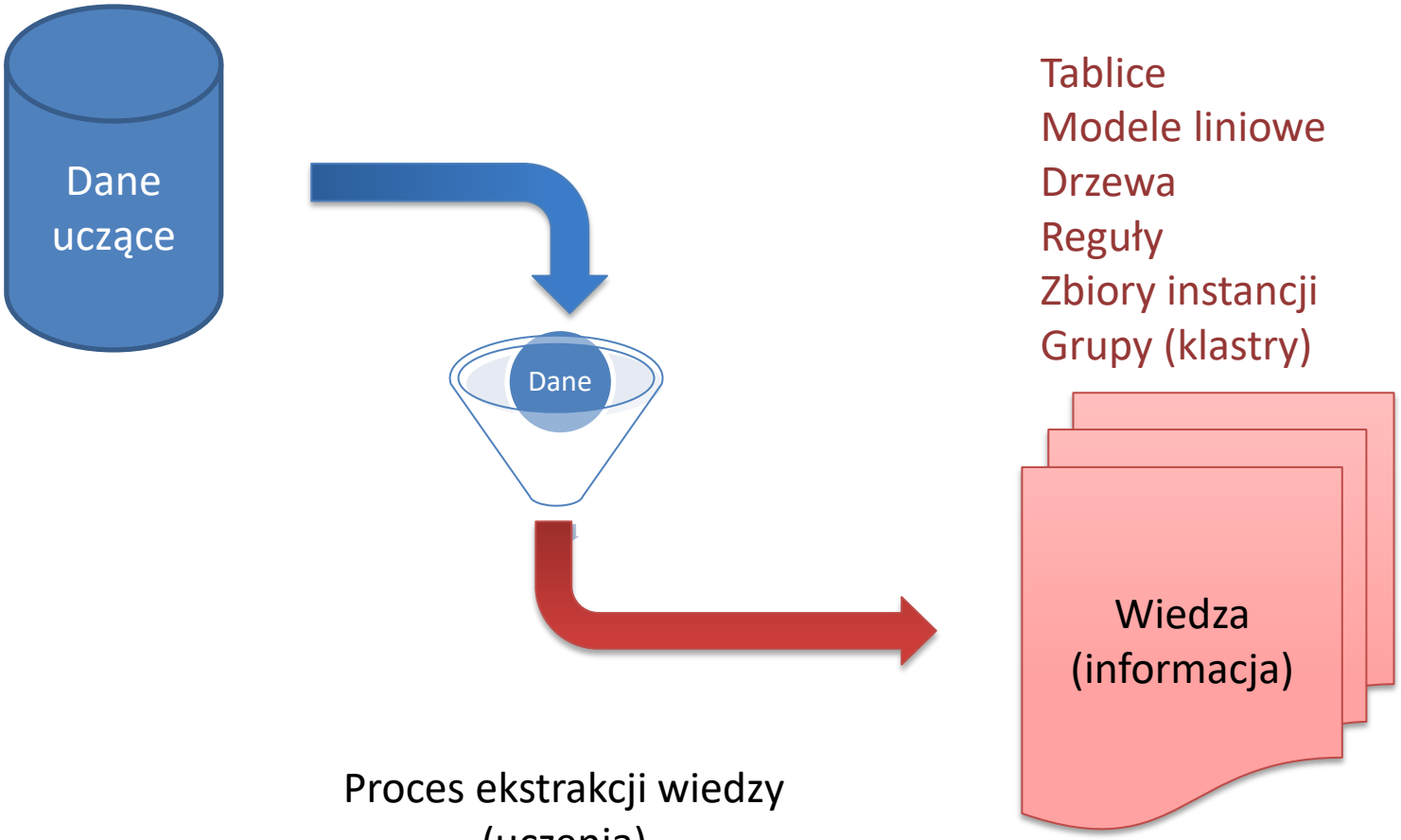
Uczenie nienadzorowane, opisowe

- Wejściem jest zbiór: $\mathrm{D} = \{x_i\}_{i=1,m}$
- Celem jest **odkrywanie wiedzy**: znalezienie interesujących wzorców wewnątrz danych, np. często występujących kombinacji atrybutów lub grup podobnych obiektów.
- Problemem jest brak jasnych kryteriów oceny

Outlook	temp	humidity	windy
sunny	85	85	FALSE
sunny	80	90	TRUE
overcast	83	86	FALSE
rainy	70	96	FALSE
rainy	68	80	FALSE
rainy	65	70	TRUE
overcast	64	65	TRUE
sunny	72	95	FALSE
sunny	69	70	FALSE
rainy	75	80	FALSE
sunny	75	70	TRUE
overcast	72	90	TRUE
overcast	81	75	FALSE
rainy	71	91	TRUE

```
Final cluster centroids:
                                Cluster#
Attribute       Full Data             0             1
                   (14.0)        (9.0)         (5.0)
=============================================================
outlook             sunny        sunny      overcast
temperature       73.5714      75.8889          69.4
humidity          81.6429      84.1111          77.2
windy               FALSE        FALSE          TRUE
```

Reprezentacja wiedzy



Tablice
Modele liniowe
Drzewa
Reguły
Zbiory instancji
Grupy (klastry)

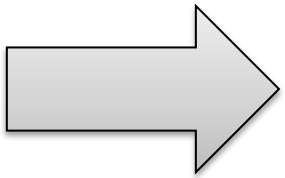
Proces ekstrakcji wiedzy (uczenia)

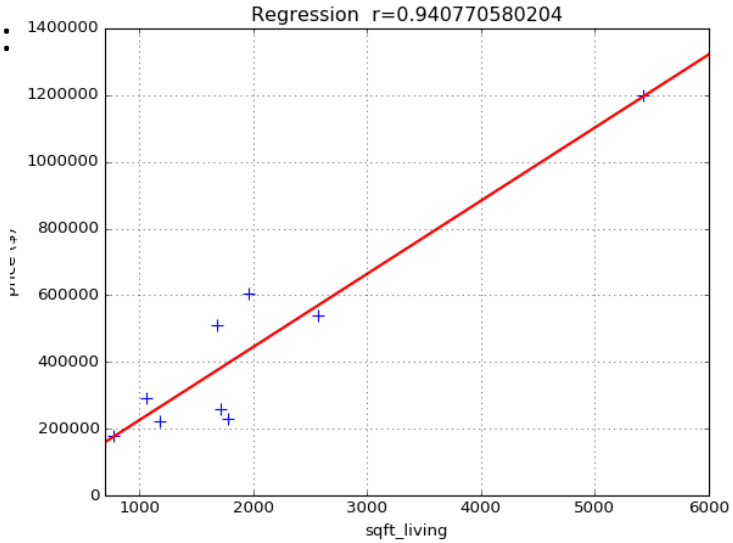
Reprezentacja wiedzy - tablica

Najprostszą formą reprezentacji wiedzy jest tablica:

- tablica decyzyjna dla klasyfikacji
- tablica regresji

sqft_living	price
770	180000
1060	291850
1180	221900
1680	510000
1715	257500
1780	229500
1960	604000
2570	538000
5420	1200000



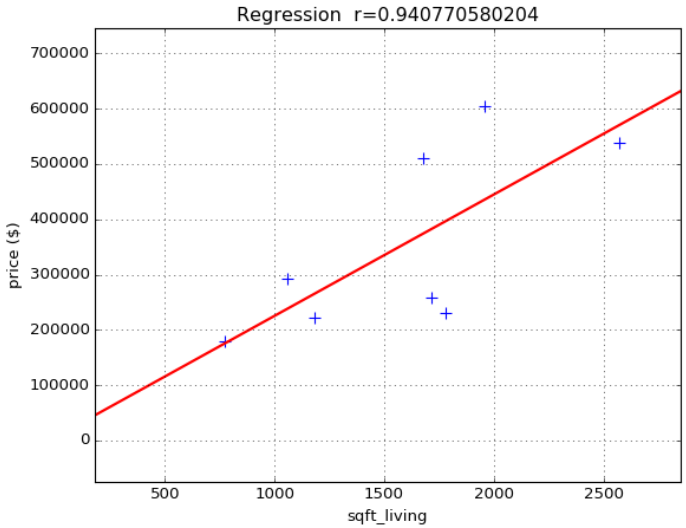


sqft_living	price
700	159 604
1289	288 792
1878	417 980
2467	547 168
3056	676 356
3644	805 544
4233	934 732
4822	1 06 3 920
5411	1 193 108
6000	1 322 296

Modele liniowe 1

Model liniowy ma postać sumy atrybutów z wagami

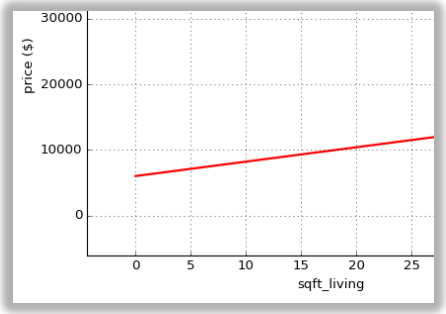
$$price = 219.38 * sqft\_living + 6040.83$$



Przesunięcie w kierunku pionowym nazywane jest w języku angielskim **intercept** lub **bias**

Model liniowy może też być wykorzystany do klasyfikacji:

- domy poniżej linii są źle sprzedane/okazyjnie kupione
- domy powyżej linii są dobrze sprzedane/przepłacone

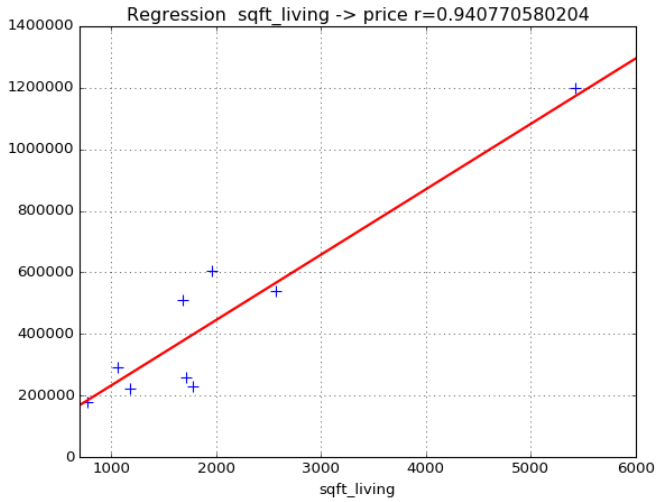


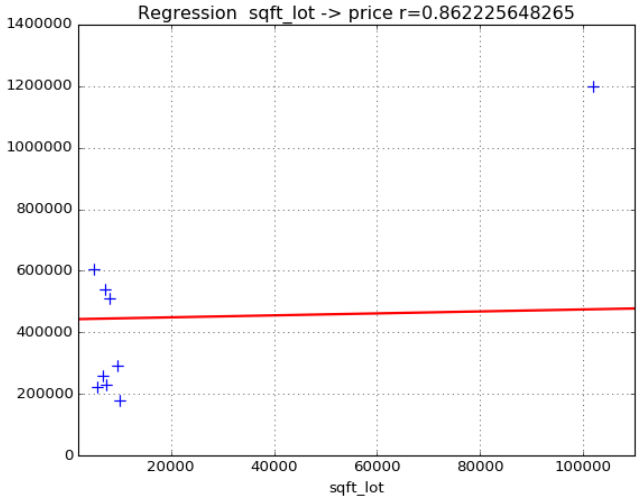
W problemach klasyfikacji linia rozgraniczająca te dwa obszary pełni rolę granicy, na której następuje zmiana decyzji. Termin angielski to **decision boundary**.

Modele liniowe 2

Model liniowy może również zostać wyznaczony dla większej liczby atrybutów

$$price = 212.73 * sqft\_living + 0.32 * sqft\_lot + 13662.28$$





Wykresy zależności ceny dla średnich wartości drugiej zmiennej.

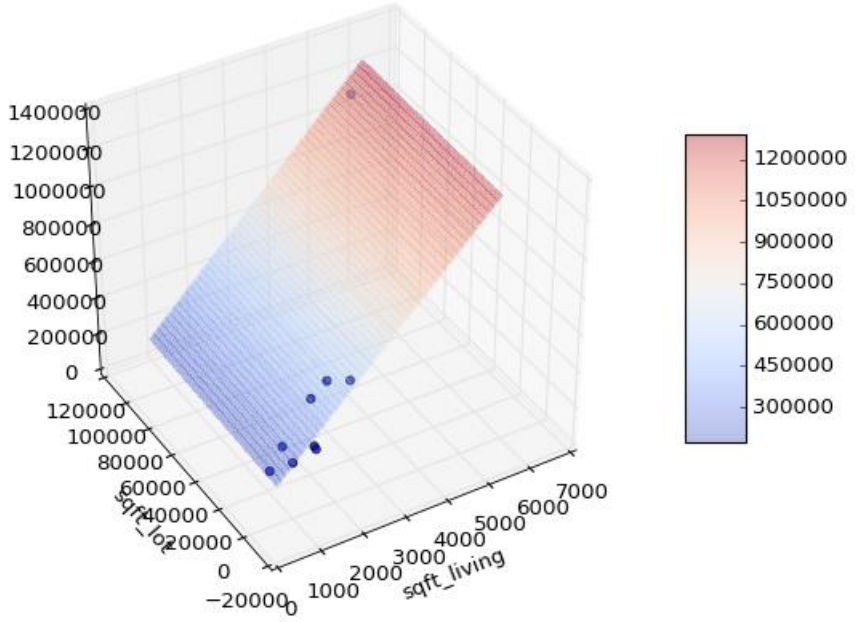
Modele liniowe 3

Model liniowy można przedstawić także jako:

$$13662.28 + 212.73 * sqft\_living + 0.32 * sqft\_lot - price = 0$$

lub jako: $[13662.28 \quad 212.73 \quad 0.32 \quad -1] \begin{bmatrix} 1 \\ sqft\_living \\ sqft\_lot \\ price \end{bmatrix} = 0$

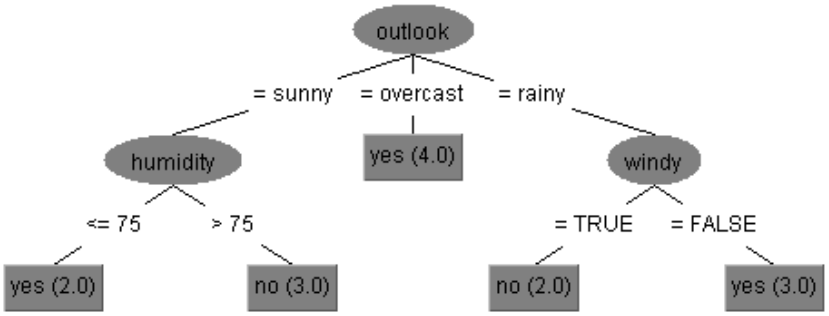
Równanie postaci $w^T h = 0$ jest równaniem hiperpłaszczyzny



Drzewa

- Drzewa najczęściej są stosowane w klasyfikacji.
- Liściom przypisana jest decyzja (etykieta klasy)
- Każdy wierzchołek drzewa jest związany z testem wartości pewnego atrybutu
  - Jeśli testowany jest atrybut nominalny, wówczas liczba węzłów potomnych odpowiada liczbie możliwych wartości
  - Jeśli testowany jest atrybut numeryczny, zwykle test polega na porównaniu ze stałą. Atrybuty numeryczne mogą być testowane wielokrotnie

outlook	temp	humidity	windy	play
sunny	85	85	FALSE	no
sunny	80	90	TRUE	no
overcast	83	86	FALSE	yes
rainy	70	96	FALSE	yes
rainy	68	80	FALSE	yes
rainy	65	70	TRUE	no
overcast	64	65	TRUE	yes
sunny	72	95	FALSE	no
sunny	69	70	FALSE	yes
rainy	75	80	FALSE	yes
sunny	75	70	TRUE	yes
overcast	72	90	TRUE	yes
overcast	81	75	FALSE	yes
rainy	71	91	TRUE	no

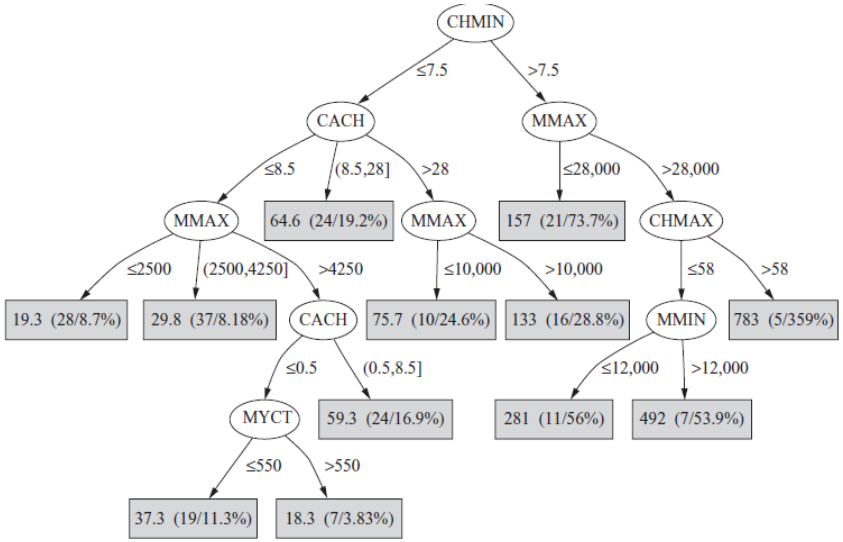


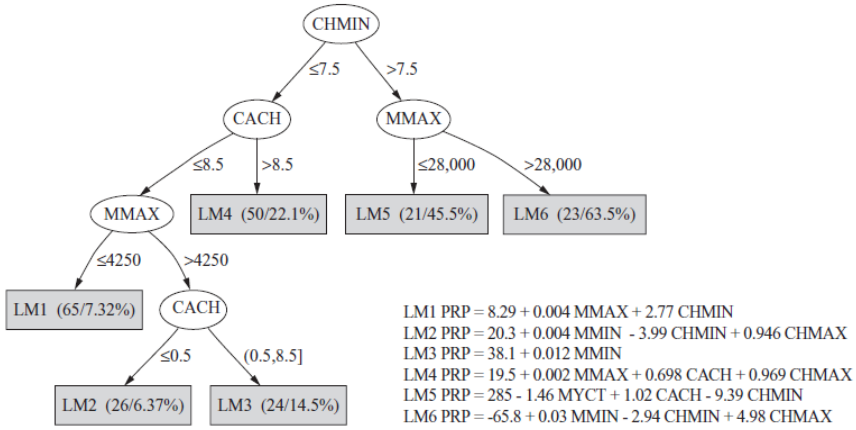
W nawiasach podano liczby instancji spełniających kryteria na ścieżce od korzenia do liścia

Nie wszystkie instancje muszą być sklasyfikowane poprawnie !

Drzewa

- Drzewa mogą być także używane do przewidywania wartości numerycznych. Każdemu liściowi przypisany jest zbiór instancji zbioru uczącego.
  - Drzewo regresji (*regression tree*) - liść przechowuje wartość średnią atrybutu wyjściowego
  - Drzewo modeli (*model tree*) – liściowi przypisane jest równanie liniowe





Ian H. Witten, Eibe Frank, Mark A. Hall Data Mining Practical Machine Learning Tools and Techniques

Reguły

Reguły mają ogólną konstrukcję:

$$przesłanki \rightarrow konkluzje$$

Zazwyczaj $przesłanki$ są koniunkcją pewnej liczby warunków, stąd typowa postać reguły to:

$$R\colon\ p_1 \wedge p_2\ \ldots p_k \rightarrow q$$

Termy p_1, p_2, p_k, q to porównania wartości atrybutu ze stałą, np.

$$p_1 \equiv outlook = sunny$$
$$\mathrm{q} \equiv play = no$$

Reguły decyzyjne (stosowane do klasyfikacji) - konkluzją reguły jest przypisanie etykiety klasy.

Reguły decyzyjne

Przykład (rekomendacja soczewek kontaktowych)

Relation: contact-lenses

No.	1: age Nominal	2: spectacle-prescrip Nominal	3: astigmatism Nominal	4: tear-prod-rate Nominal	5: contact Nomin
1	young	myope	no	reduced	none
2	young	myope	no	normal	soft
3	young	myope	yes	reduced	none
4	young	myope	yes	normal	hard
5	young	hypermetrope	no	reduced	none
6	young	hypermetrope	no	normal	soft
7	young	hypermetrope	yes	reduced	none
8	young	hypermetrope	yes	normal	hard
9	pre-presbyopic	myope	no	reduced	none
10	pre-presbyopic	myope	no	normal	soft
11	pre-presbyopic	myope	yes	reduced	none
12	pre-presbyopic	myope	yes	normal	hard
13	pre-presbyopic	hypermetrope	no	reduced	none
14	pre-presbyopic	hypermetrope	no	normal	soft
15	pre-presbyopic	hypermetrope	yes	reduced	none
16	pre-presbyopic	hypermetrope	yes	normal	none
17	presbyopic	myope	no	reduced	none
18	presbyopic	myope	no	normal	none
19	presbyopic	myope	yes	reduced	none
20	presbyopic	myope	yes	normal	hard
21	presbyopic	hypermetrope	no	reduced	none
22	presbyopic	hypermetrope	no	normal	soft
23	presbyopic	hypermetrope	yes	reduced	none

presbyopic – dalekowzroczność związana z wiekiem
myope – krótkowidz
hypermetrope - dalekowidz

(tear-prod-rate = normal) and (astigmatism = yes) => contact-lenses=hard (6.0/2.0)
(tear-prod-rate = normal) => contact-lenses=soft (6.0/1.0)
=> contact-lenses=none (12.0/0.0)

Reguły decyzyjne

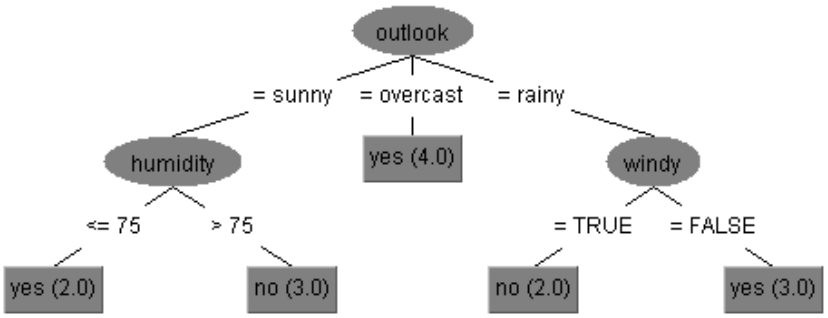
(tear-prod-rate = normal) and (astigmatism = yes) => contact-lenses=hard (6.0/2.0)
(tear-prod-rate = normal) => contact-lenses=soft (6.0/1.0)
=> contact-lenses=none (12.0/0.0)

Problemy:

- dla danej instancji mogą być prawdziwe przesłanki różnych reguł
  - uporządkowana lista czy zbiór wykonywany w dowolnej kolejności
  - priorytet reguły, np. wybierana jest reguła, która była częściej odpalona w zbiorze uczącym
- dla danej instancji w zbiorze $\{R_i\}$ może nie istnieć spełniona reguła
  - reguła standardowa (o najniższym priorytecie) contact-lenses=none (12.0/0.0)

Reguły decyzyjne

Reguły mogą być łatwo wyznaczone na podstawie drzewa decyzyjnego



(outlook = sunny) and (humidity<=75) => play=yes
(outlook = sunny) and (humidity<=75) => play=no
(outlook = overcast) => play=yes
(outlook = sunny) and (windy=TRUE) => play=no
(outlook = sunny) and (windy=FALSE) => play=no

Jednakże algorytmy przeznaczone do generacji reguł są (zazwyczaj) bardziej efektywne

```
JRIP rules:
===========

(humidity >= 85) and (outlook = sunny) => play=no (3.0/0.0)
(outlook = rainy) and (windy = TRUE) => play=no (2.0/0.0)
 => play=yes (9.0/0.0)

Number of Rules : 3
```

Reguły asocjacyjne

- Reguły asocjacyjne reprezentują odkryte wzorce w zbiorze danych, czyli statystycznie istotne kombinacje atrybutów.
  - **Wsparcie** (*support/coverage*) - liczba instancji, dla których przesłanka reguły jest prawdziwa
  - **Ufność** (*confidence/accuracy*) – liczba instancji, dla których wartość określona w konkluzji jest przewidywana poprawnie
- Reguły asocjacyjnie są traktowane całkowicie niezależnie i nie mają bezpośredniego zastosowania do klasyfikacji

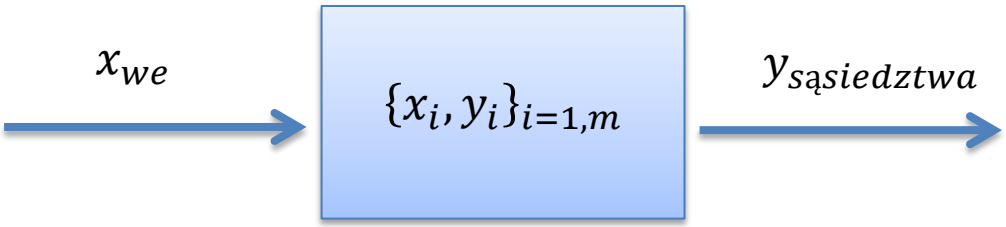
10 najlepszych reguł odkrytych przez algorytm Apriori.

Algorytm działa wyłącznie na danych kategorycznych, dlatego wartości numeryczne zostały zastąpione symbolami przedziałów, np. temperature={hot, mild, cool}

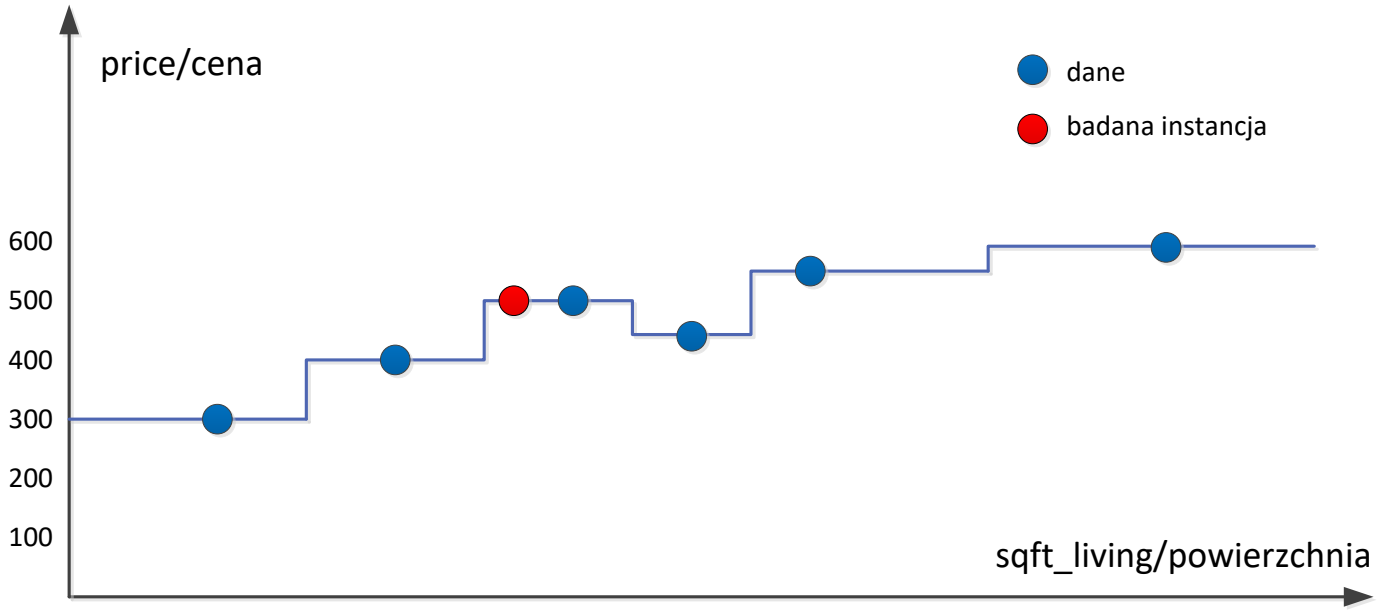
1. **outlook=overcast 4 ==> play=yes 4** <conf:(1)> lift:(1.56) lev:(0.1) [1] conv:(1.43)
2. **temperature=cool 4 ==> humidity=normal 4** <conf:(1)> lift:(2) lev:(0.14) [2] conv:(2)
3. **humidity=normal windy=FALSE 4 ==> play=yes 4** <conf:(1)> lift:(1.56) lev:(0.1) [1] conv:(1.43)
4. outlook=sunny play=no 3 ==> humidity=high 3 <conf:(1)> lift:(2) lev:(0.11) [1] conv:(1.5)
5. **outlook=sunny humidity=high 3 ==> play=no 3** <conf:(1)> lift:(2.8) lev:(0.14) [1] conv:(1.93)
6. outlook=rainy play=yes 3 ==> windy=FALSE 3 <conf:(1)> lift:(1.75) lev:(0.09) [1] conv:(1.29)
7. **outlook=rainy windy=FALSE 3 ==> play=yes 3** <conf:(1)> lift:(1.56) lev:(0.08) [1] conv:(1.07)
8. temperature=cool play=yes 3 ==> humidity=normal 3 <conf:(1)> lift:(2) lev:(0.11) [1] conv:(1.5)
9. **outlook=sunny temperature=hot 2 ==> humidity=high 2** <conf:(1)> lift:(2) lev:(0.07) [1] conv:(1)
10. temperature=hot play=no 2 ==> outlook=sunny 2 <conf:(1)> lift:(2.8) lev:(0.09) [1] conv:(1.29)

Zbiór instancji

Reprezentacją wiedzy jest zbiór instancji ze znanymi wartościami wyjściowymi (wartością numeryczną lub etykietą klasy).



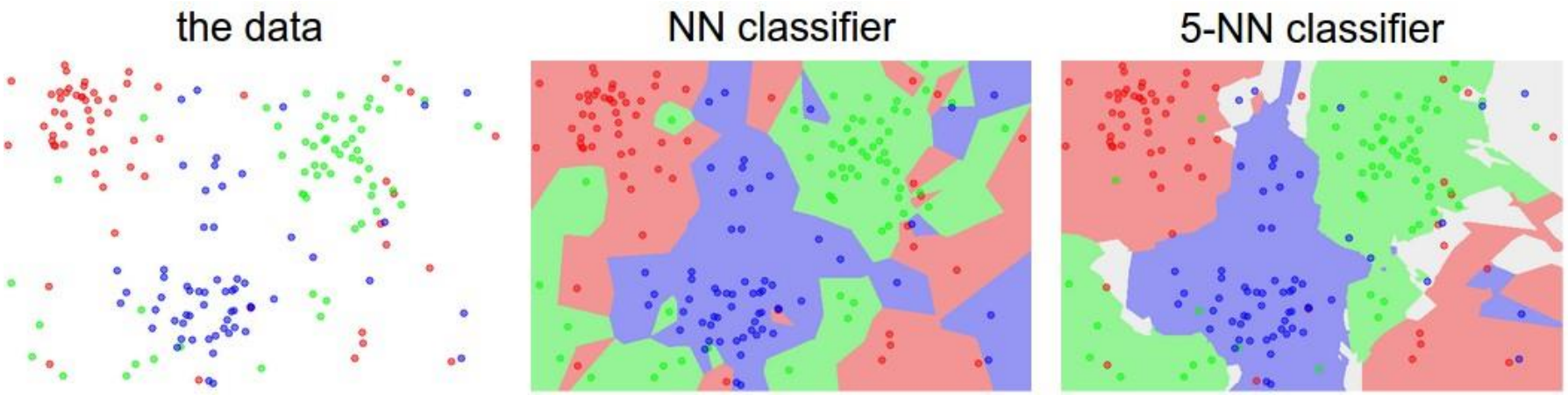
Zwracana jest wartość wyjściowa $y_{sąsiedztwa}$ wyznaczona na podstawie sąsiadujących instancji.



Zbiór instancji

W przypadku klasyfikacji najbardziej znany jest algorytm k-NN (k Nearest Neighbors):

- wyznaczanych jest k najbliższych sąsiadów
- wybrana zostanie etykieta, która przeważa w otoczeniu



[Źródło: http://cs231n.github.io/classification/]

- Za pomocą kolorów odwzorowano obszary, dla których wybrana zostanie dana etykieta
- Kształt obszarów zależy od użytej miary odległości
- Białe obszary – brak jednoznacznej klasyfikacji

Grupowanie

Wynikiem grupowania (klasteryzacji) jest przydział etykiet (numerów klastrów) do instancji.

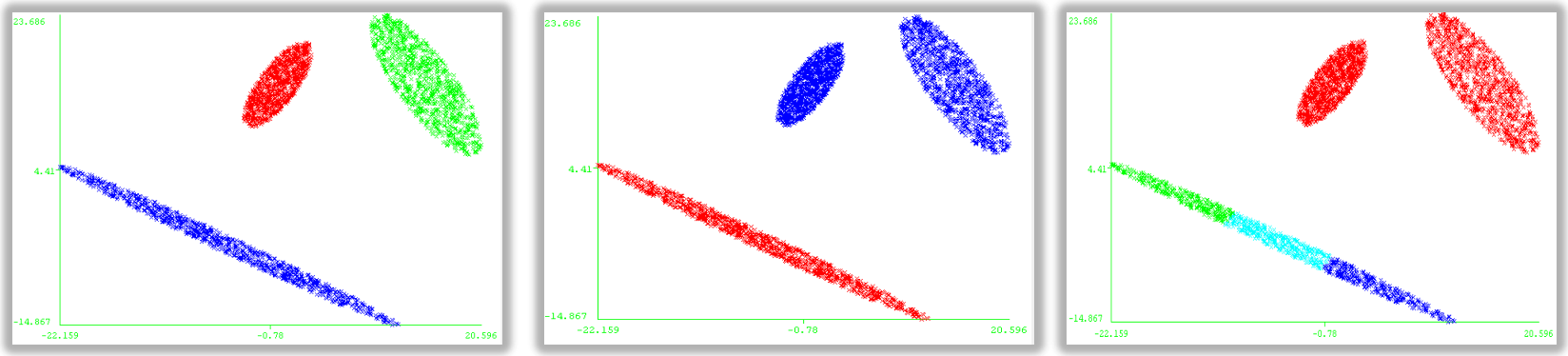
Możliwe są różne wersje grupowania:

- pojedynczą etykietą: $x_i \rightarrow L$ (rozłączne)
- zbiorem etykiet $x_i \rightarrow 2^L$, czyli instancja należy do kilku grup. Zazwyczaj jest to funkcja $x_i \rightarrow 2^{L \times R}$. Dodatkowym parametrem jest waga określająca stopień przynależności do grupy,

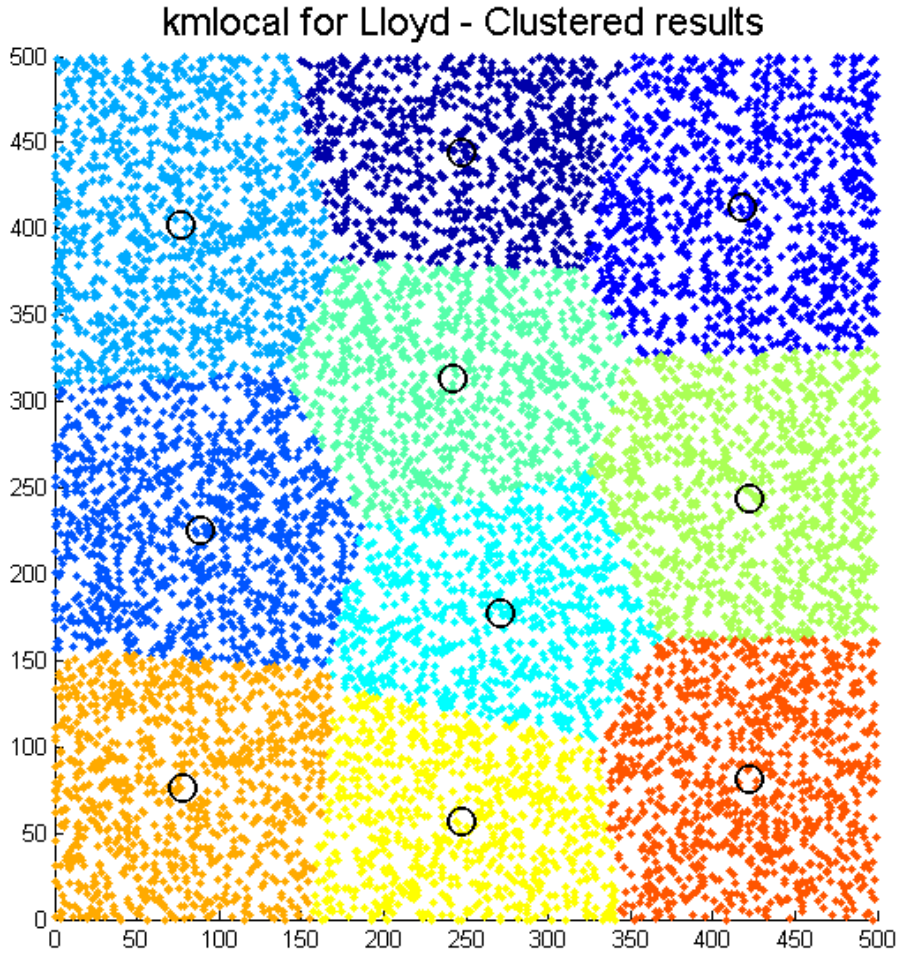
$$film_i \rightarrow \{(Fantasy, 0.6), (Drama, 0.2), (Thriller, 0.2)\}$$

- w przypadku hierarchicznego grupowania etykiety dodatkowo są ułożone w hierarchię

Przykład: wynik działania różnych algorytmów grupowania rozłącznego



Grupowanie rozłączne

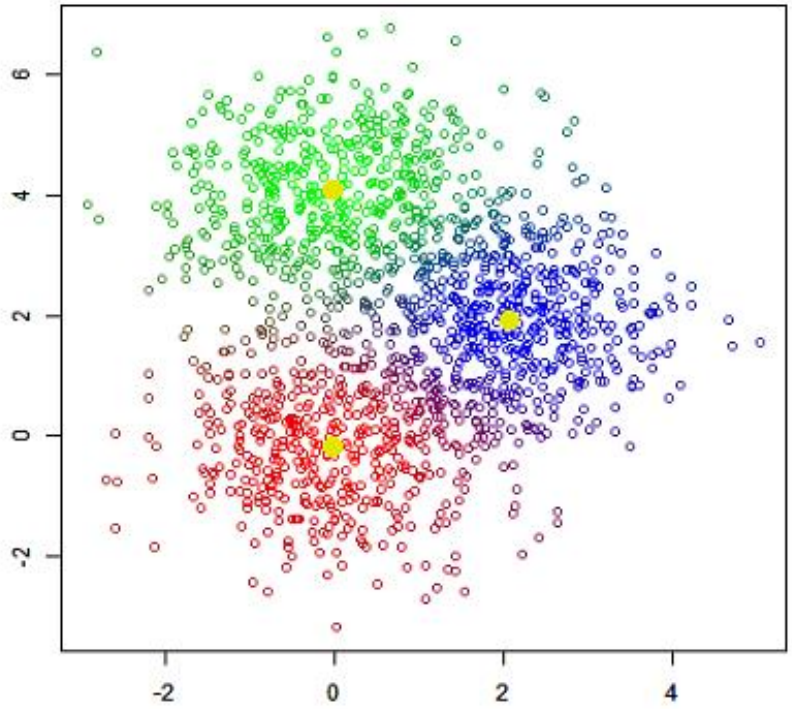


Prawdopodobnie najbardziej znanym algorytmem grupowania rozłącznego jest k-means.

Wyznacza on środki grup i tym samym podział na obszary należące do grup skupionych wokół środka.

[https://summerofhpc.prace-ri.eu/quizz-clustered-data-using-k-means/]

Grupowanie nierozłączne

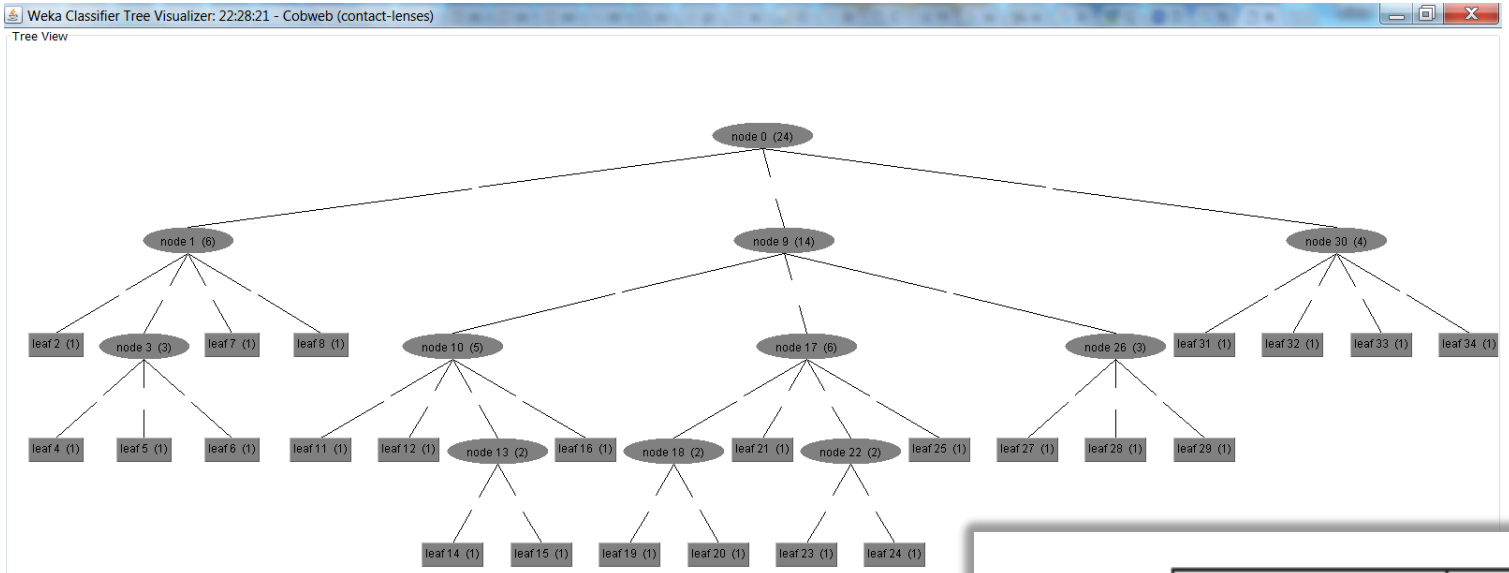


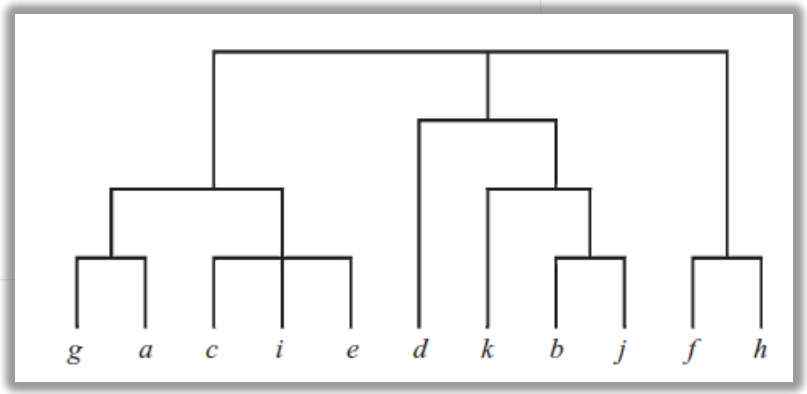
Typowym algorytmem grupowania nierozłącznego jest fuzzy c-means

[http://fistofmath.sakura.ne.jp/]

Grupowanie hierarchiczne

- W grupowaniu hierarchicznym etykiety zorganizowane są w hierarchie.
- Wynik prezentowany jest w postaci tzw. **dendrogramu**.





Koncept

- Koncept (ang. *concept*) jest dość niejasnym terminem
- W przybliżeniu oznacza **model wraz z parametrami** będący wynikiem algorytmu uczenia
- Konceptem może być:
  - funkcja regresji
  - klasyfikator, np. w postaci zbioru reguł
  - sposób podziału na obiektów na grupy, wtedy np.: koncept jest reprezentowany przez zbiór wektorów definiujących ich punkty środkowe
- Tradycyjnie, **koncept = pojęcie = klasa**, czyli specyfikacja zbioru obiektów posiadających wspólne cechy
- Ważnym pojęciem jest dryft modelu (*concept drift*)
  - w przypadku przetwarzania strumienia danych wyznaczony wcześniej model przestaje być adekwatny, np. ceny domów rosną, użytkownicy zmieniają zachowanie i wybierają inne towary, wyposażają domy w klimatyzację i zużywają więcej energii podczas upałów…
  - w praktycznych zastosowaniach systemy korzystające z algorytmów eksploracji danych powinny wykrywać spadek jakości modeli i przeprowadzić powtórny proces uczenia

Oprogramowanie

Praktycznie każdy dostawca oprogramowania dla baz danych/hurtowni danych zapewnia zintegrowane narzędzia eksploracji danych.

Dostępnych jest dużo pakietów oprogramowania dla uczenia maszynowego

Top Data Mining Software

IBM SPSS Modeler, SAS Enterprise Miner, RapidMiner, Angoss Knowledge STUDIO, Microsoft Analysis Services, Oracle Data Mining, FICO Data Management Integration Platform, Think Analytics, Salford Systems, Viscovery, Portrait Software, IBM DB2 Intelligent Miner, STATISTICA Data Miner, QIWare, LIONsolver, KXEN Modeler, Neural Designer, Megaputer's PolyAnalyst, TIBCO Spotfire Miner, XLMiner- Frontline Systems, GhostMiner, Teradata Warehouse Miner, KNIME, Advanced Miner, Alteryx Designer and Rapid Insight Veera

Top Free Data Mining Software

Orange, Weka,Rattle GUI, Apache Mahout, SCaViS, RapidMiner, R, ML-Flex, Databionic ESOM Tools, Natural Language Toolkit, SenticNet API , ELKI , UIMA, KNIME, Chemicalize.org , Vowpal Wabbit, GNU Octave, CMSR Data Miner, Mlpy, MALLET, Shogun, Scikit-learn, LIBSVM, LIBLINEAR, Lattice Miner, Dlib, Jubatus, KEEL, Gnome-datamine-tools, Alteryx Project Edition , OpenNN, ADaM, ROSETTA, ADaMSoft, Anaconda, yooreeka, AstroML, streamDM, jHepWork, TraMineR, ARMiner, arules, CLUTO and TANAGRA.

[http://www.predictiveanalyticstoday.com/top-data-mining-software/]

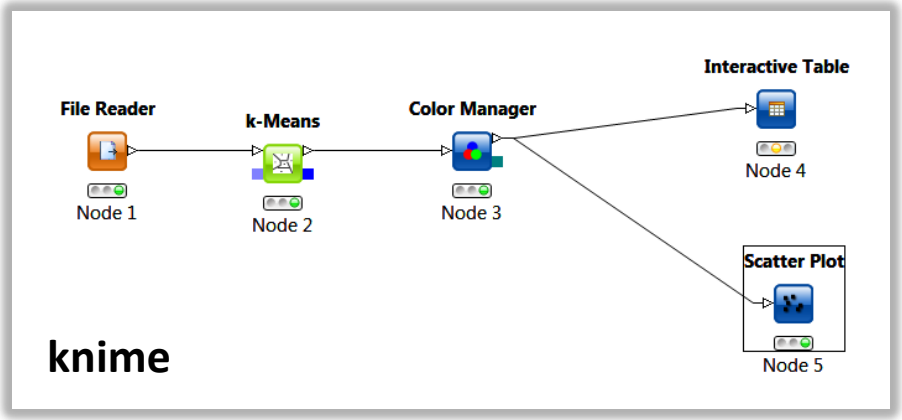
Języki i środowiska

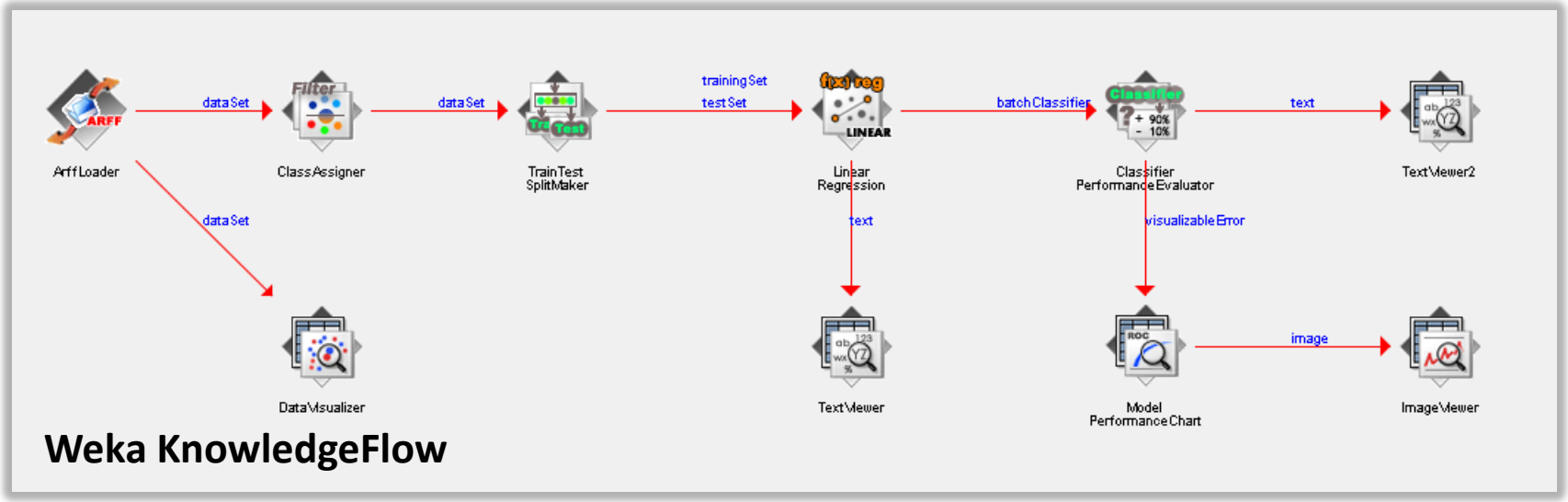
- Matlab/Octave
- Java: Weka, JavaML
- Python: SciPy, scikit-learn, Pandas, Graphlab Create
- R https://www.r-project.org/
- Julia: „designed for Parallelism and Cloud Computing”

- Apache Spark (Java, Scala, Python, R) http://spark.apache.org/
- Apache Mahout https://mahout.apache.org/
- MOA (Java, Weka) http://moa.cms.waikato.ac.nz/

Środowiska typu workflow

- Większość narzędzi dla analiz biznesowych
- RapidMiner
- knime
- Weka

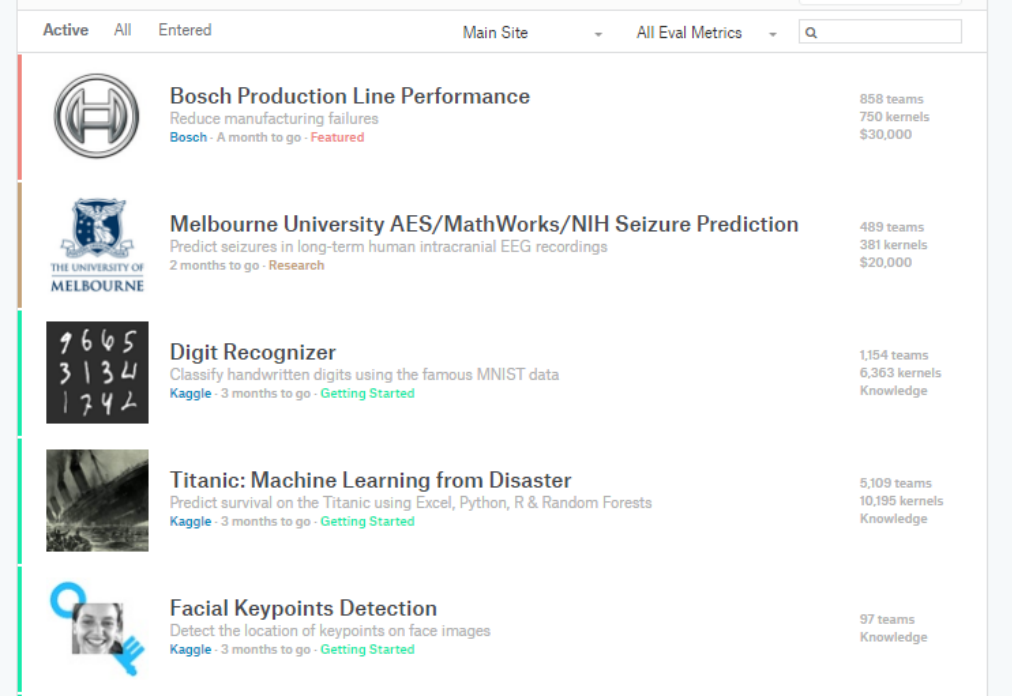




kaggle 1

Społeczność skupiona wokół eksploracji danych: https://www.kaggle.com/

- zawody (competitions)
- zbiory danych (datasets)
- programy (kernels): Python, R, Julia, SQLite oraz platforma ich wykonania w chmurze
- oferty pracy



kaggle 2

Zbiory danych (2016)