

# Machine Learning

## Systemy uczące się

### w1: wprowadzenie



**Polish-SIGML**

**Jerzy Stefanowski**  
Instytut Informatyki  
Politechnika Poznańska

Poznań, 2015/16  
aktualizacja 2020

# Uwagi organizacyjne

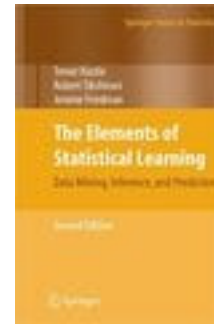


- **Wykłady** – dr hab. inż. Jerzy Stefanowski, prof. PP
- **Laboratorium** – prowadzący: dr hab. inż. M. Komosiński (ISWD), mgr inż. Mateusz Lango i Konrad Miazga (ITI)
- Książki
  - Systemy uczące się. Cichosz P., WNT, 2000.
  - Uczenie maszynowe i sieci neuronowe. Krawiec K., Stefanowski J., Wyd. PP 2004
  - Sztuczne sieci neuronowe. Żurada J., Barski M., Jędruch W., PWN 1996
  - Bardzo dobre książki w języku angielskim!!!
  - Tzw. kursy online (język ang. i wybierz najlepsze uniwersytety amerykańskie)
- Zasady zaliczenia – test pisemny
- Udostępnianie materiałów i [zWWW](#) repozytorium

# Podręczniki anglojęzyczne

---

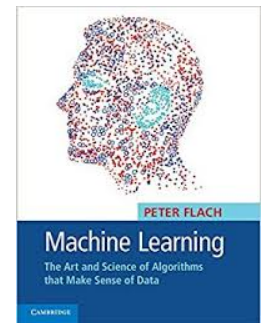
Elements of statistical learning (HTF)



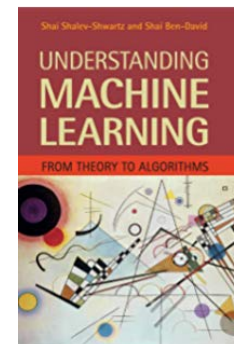
Pattern Recognition and Machine Learning (Ch Bishop)



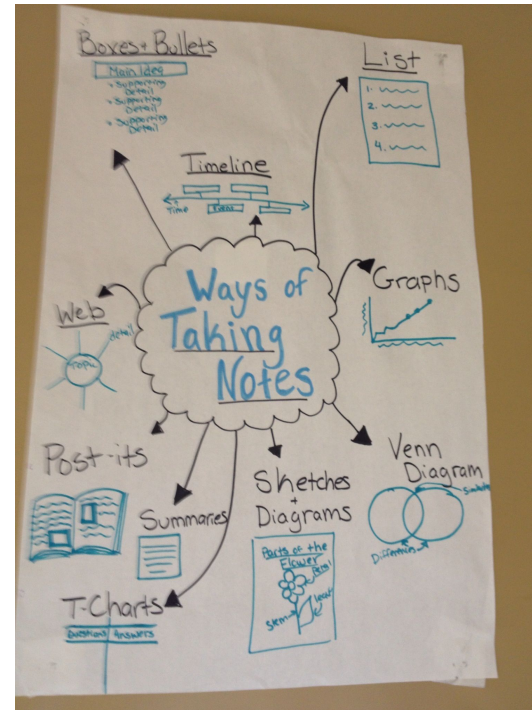
Machine learning: The Art and Science of Algorithms that Make Sense of Data. (Peter A. Flach)



Understanding Machine Learning (S. Ben-David, S, Shalev-Shwartz )

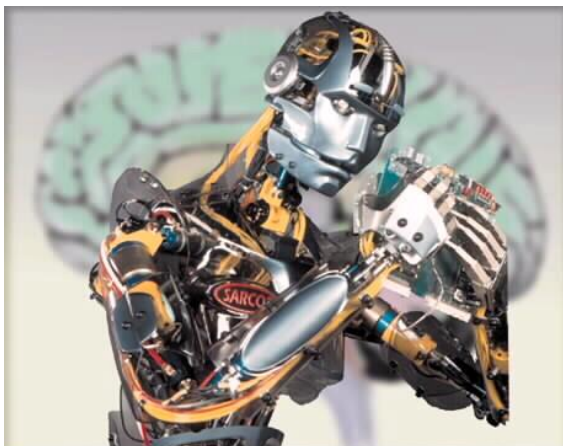


# Niezależnie należy prowadzić notatki!!!!

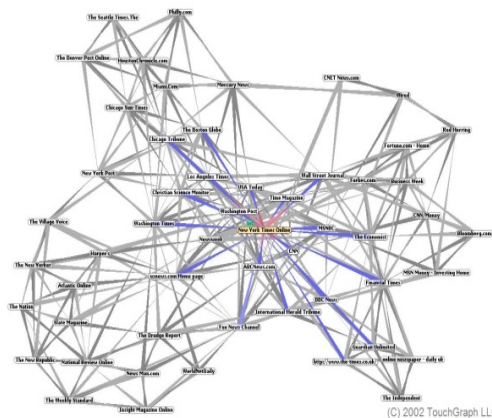
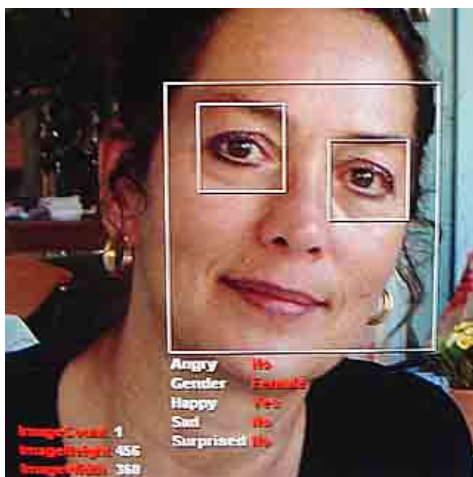


Czym jest wykład na uniwersytecie?  
Czy slajdy zawierają pełne materiały?

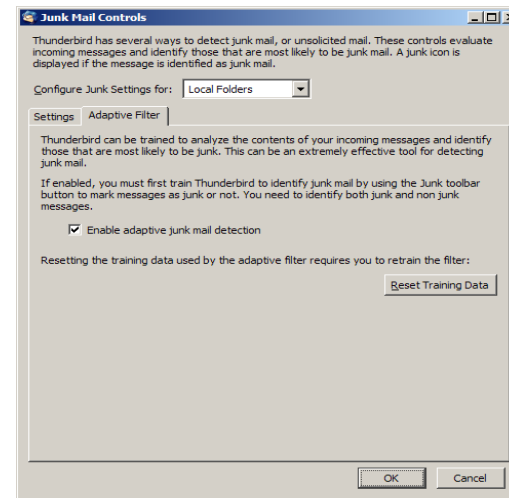
# Gdzie odnaleźć systemy uczące się?



© Can Stock Photo - csp21075804



(C) 2002 TouchGraph LLC



# Oferty pracy (od kilku lat)

## INŻYNIER - UCZENIE MASZYNOWE I ROZPOZNAWANIE WZORCÓW

Miejsce pracy: **Warszawa**  
Nr Ref.: **SSD/AI/MPR**

### Zakres odpowiedzialności

- o Udział w projektach związanych z przetwarzaniem i rozpoznawaniem biosygnatów oraz sygnałów audio i wideo
- o Projektowanie oraz implementacja nowych algorytmów oraz testowanie i ulepszanie istniejących
- o Współpraca z innymi jednostkami R&D i uczelniami

### Wymagania

- o Wykształcenie wyższe o specjalności informatyka, matematyka stosowana, elektronika, telekomunikacja, multimedia lub pokrewnej
- o Co najmniej 3-letnie doświadczenie w programowaniu w języku C++ oraz Matlab lub Python
- o Doświadczenie w projektowaniu oraz implementacji algorytmów uczenia maszynowego i rozpoznawania wzorców
- o Umiejętność szybkiego rozwiązywania problemów
- o Dobra znajomość języka angielskiego w mowie i piśmie

### Kwalifikacje będące dodatkowym atutem

- o Doktorat w dziedzinie powiązanej z uczeniem maszynowym/rozpoznawaniem wzorców

### Data Scientist Intern - Advanced Analytics

McKinsey & Company - ★★★★★ 329 ocen - Warszawa, mazowieckie  
Data preparation skills and working knowledge of SQL. You will drive insights generation base on descriptive, predictive & prescriptive analytics of Big Data....  
27 dni temu - [zachowaj ofertę](#) - [więcej...](#)

### Data Scientist

Nexio Management - Warszawa, mazowieckie  
Co najmniej 4 lata doświadczenia w obszarze Data Science. Tworzenie modeli klasyfikacyjnych lub predykcyjnych za pomocą metod i algorytmów uczenia maszynowego....  
Absolvent.pl - 3 dni temu - [zachowaj ofertę](#) - [więcej...](#)

### Statistician / Data Scientist

Citi Service Center Poland - Warszawa, mazowieckie  
Statistician / Data Scientist. Detect data anomalies and identify source of data problems. Generate graphical data reports, including data cross-tables, scatter...  
Oferta sponsorowana przez [careersinpoland.com](#) - [zachowaj ofertę](#)

### Data Mining Expert/Data Scientist - Model Development Team

Citi Service Center Poland - Warszawa, mazowieckie  
Data Mining Expert/Data Scientist - Model Development Team. Detect data anomalies and identify source of data problems. Recommend data quality solutions....

## Analitik ds. Machine Learning

### Kogo szukamy?

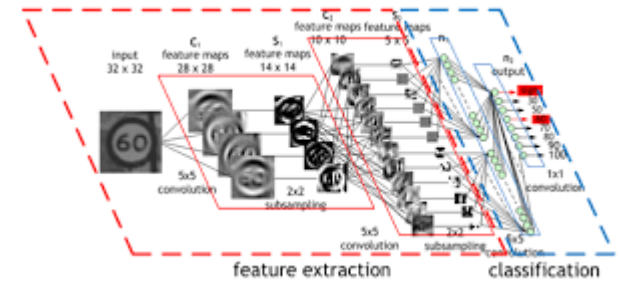
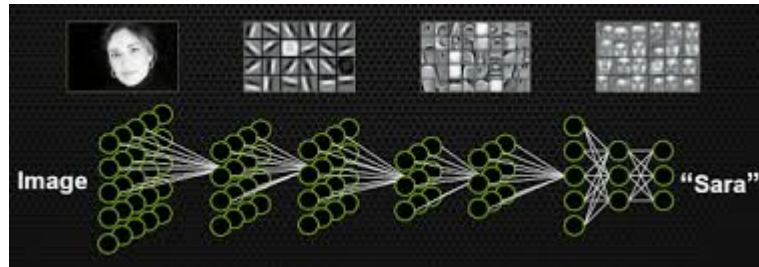
Poszukujemy osoby z pasją i nieszablonywym podejściem do realizacji zadań, która dotyczy do zespołu wspierającego biznes zaawansowanymi rozwiązaniami analitycznymi. Obszar zadań będzie związany z wykorzystaniem modeli statystycznych i sztucznej inteligencji w celu usprawnienia procesów obsługi klienta.

### Czego oczekujemy?

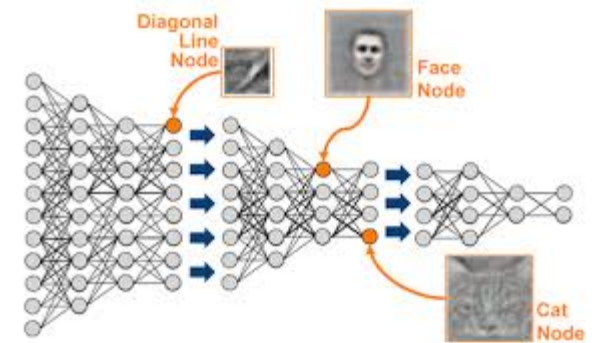
- o ukończonych studiów wyższych na kierunku: statystyka, matematyka, informatyka lub pokrewne
- o nastawienia na ciągły rozwój i poszukiwanie nowych rozwiązań
- o dobrej znajomości algorytmów uczenia maszynowego
- o dobrej znajomości narzędzi: R, SQL,Python
- o doświadczenia zawodowego lub naukowego w obszarze przetwarzania danych nieustrukturyzowanych i analizy tekstu

Wybrane luty 2018

# Deep Neural Networks – hot topic

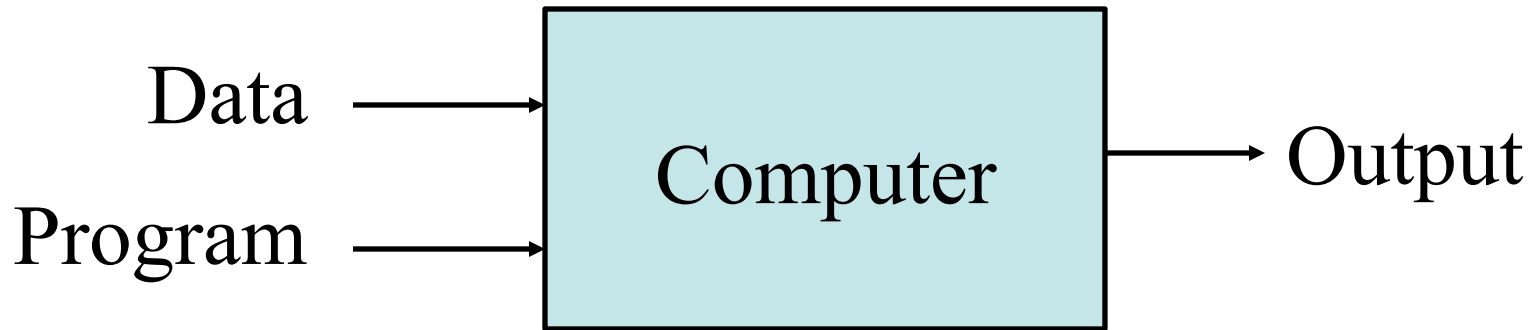


- Specjalne sztuczne sieci neuronowe
  - Rozpoznawania obrazów
  - Przetwarzanie sygnałów (np.. mowa)
  - Gry (Google DeepMind – AlphaGO)
- Lecz nie na tym przedmiocie (K.Krawiec)

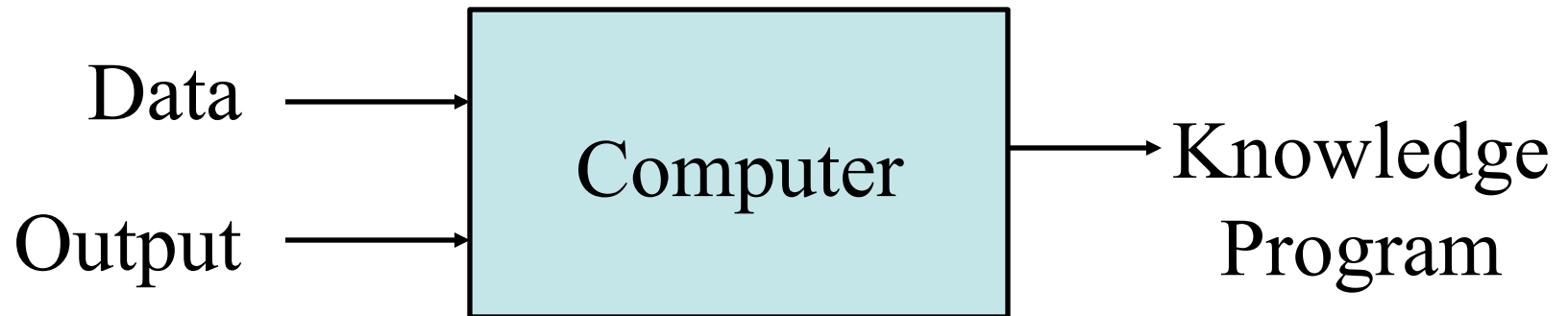


# Czym jest uczenie maszynowe?

- Traditional Approaches



- Machine Learning





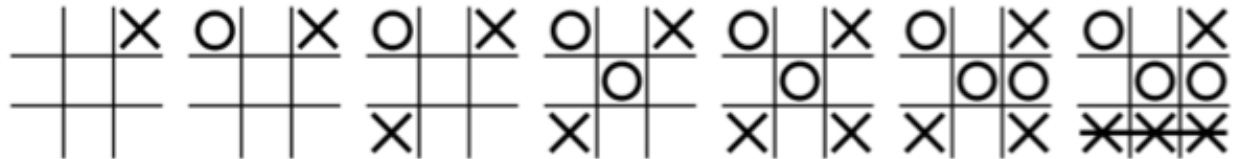
# Uczenia maszynowe / systemy uczące się

- **Machine** → program / algorytm
- **Learning** → “improving performance on a given task based on experience / examples”

Innymi słowami:

- “instead of the programmer writing explicit rules for how to solve a given problem, the programmer instructs the computer how to learn from examples”<sub>9</sub> [Kivinen, Helsinki]

- ▶ How to program the computer to play tic-tac-toe?



- ▶ Option A: The programmer writes explicit rules, e.g. 'if the opponent has two in a row, and the third is free, stop it by placing your mark there', etc (lots of work, difficult, not at all scalable!)
- ▶ Option B: Go through the game tree, choose optimally (for non-trivial games, must be combined with some heuristics to restrict tree size)
- ▶ Option C: Let the computer try out various strategies by playing against itself and others, and noting which strategies lead to winning and which to losing (= 'machine learning')

## Example 2: spam filter

- ▶ Programmer writes rules: “If it contains ‘viagra’ then it is spam.” (difficult, not user-adaptive)
- ▶ The user marks which mails are spam, which are legit, and the computer learns itself what words are predictive

<pre>From: medshop@spam.com Subject: viagra cheap meds...  From: my.professor@helsinki.fi Subject: important information here's how to ace the test...  :</pre>	<pre>spam  non-spam  :</pre>
<pre>From: mike@example.org Subject: you need to see this how to win \$1,000,000...</pre>	<pre>?</pre>

# Studium przypadku: rozpoznawanie spamu

---

## Inspiracja P.Flach:

- SpamAssassin is a widely used open-source spam filter. It calculates a score for an incoming e-mail, based on a number of built-in rules or 'tests' in SpamAssassin's terminology, and adds a 'junk' flag and a summary report to the e-mail's headers if the score is 5 or more.

-0.1 RCVD\_IN\_MXRATE\_WL RBL: MXRate recommends allowing [123.45.6.789 listed in sub.mxrate.net]

0.6 HTML\_IMAGE\_RATIO\_02 BODY: HTML has a low ratio of text to image area

1.2 TVD\_FW\_GRAPHIC\_NAME\_MID BODY: TVD\_FW\_GRAPHIC\_NAME\_MID

0.0 HTML\_MESSAGE BODY: HTML included in message

0.6 HTML\_FONx\_FACE\_BAD BODY: HTML font face is not a word

1.4 SARE\_GIF\_ATTACH FULL: Email has a inline gif

0.1 BOUNCE\_MESSAGE MTA bounce message

0.1 ANY\_BOUNCE\_MESSAGE Message is some kind of bounce message

1.4 AWL AWL: From: address is in the auto white-list

- Sum 5.3 → test says SPAM

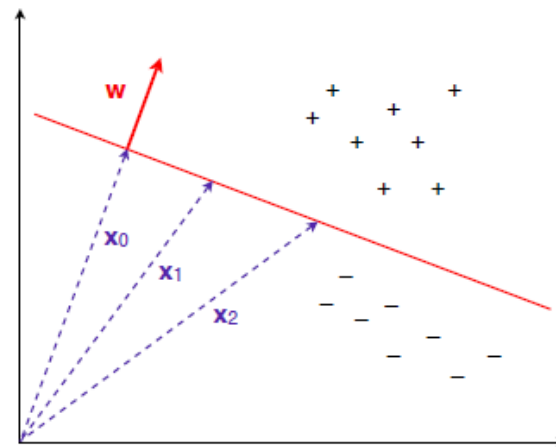
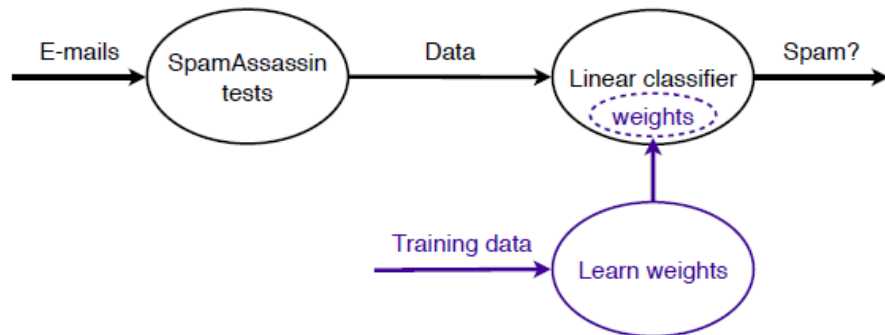
# Wykorzystanie gotowych testów

- Najprostszy model klasyfikatora liniowego

E-mail	$x_1$	$x_2$	Spam?	$4x_1 + 4x_2$
1	1	1	1	8
2	0	0	0	0
3	1	0	0	4
4	0	1	0	4

The columns marked  $x_1$  and  $x_2$  indicate the results of two tests on four different e-mails. The fourth column indicates which of the e-mails are spam. The right-most column demonstrates that by thresholding the function  $4x_1 + 4x_2$  at 5, we can separate spam from ham.

## Optimizing weights in a linear classifier



# Model klasyfikatora probabilistycznego

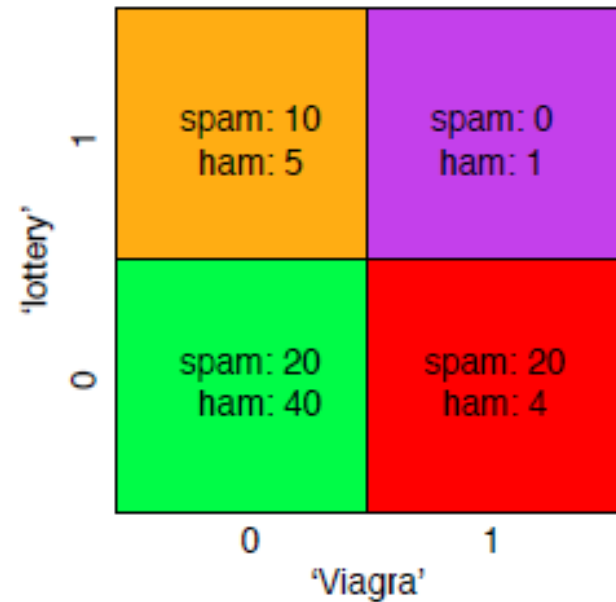
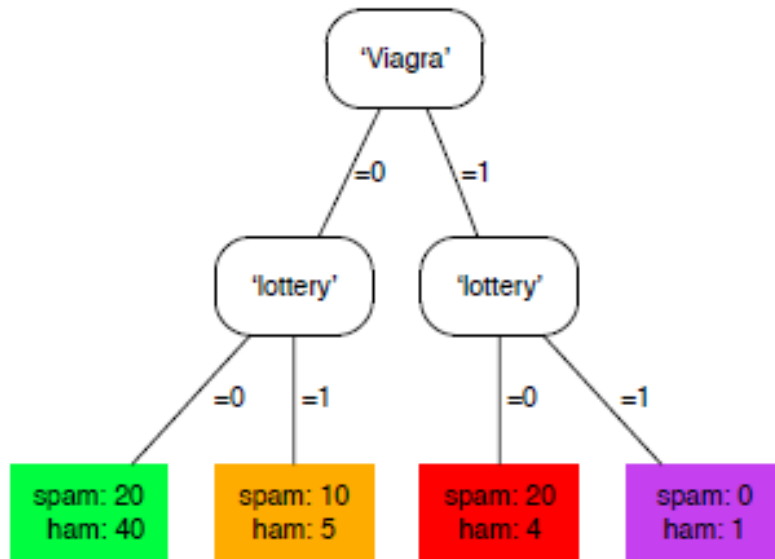
- Inne cechy – termy i ich prawdopodobieństwo wystąpienia
- Naucz Naive Bayes Classifier

Viagra	lottery	$P(Y = \text{spam} \text{Viagra}, \text{lottery})$	$P(Y = \text{ham} \text{Viagra}, \text{lottery})$
0	0	0.31	<b>0.69</b>
0	1	<b>0.65</b>	0.35
1	0	<b>0.80</b>	0.20
1	1	0.40	<b>0.60</b>

'Viagra' and 'lottery' are two Boolean features;  $Y$  is the class variable, with values 'spam' and 'ham'. In each row the most likely class is indicated in bold.

# Klasyfikatory symboliczne

- Nowe cechy – naucz drzewa albo reguły



# Dlaczego takie systemy .....

---

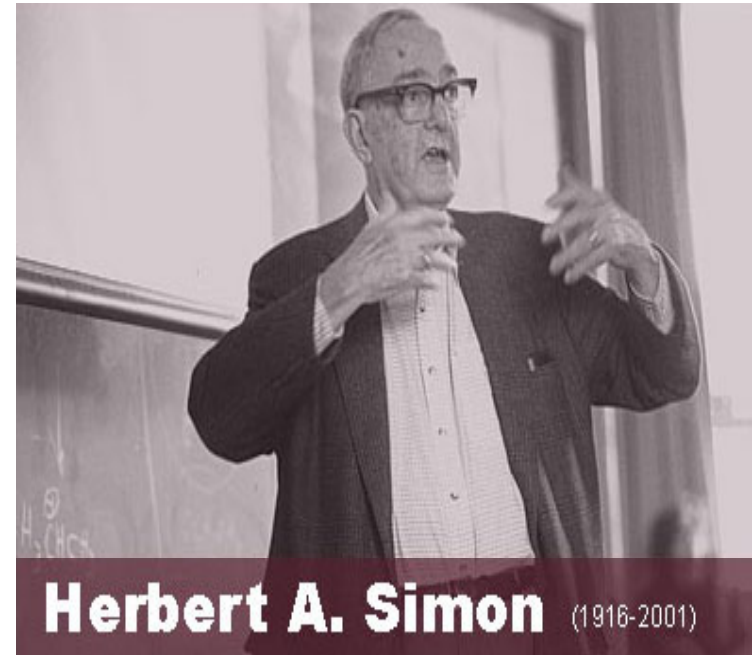
- Problemy, które są złożone, trudne do opisu i często nie posiadają wystarczających modeli teoretycznych albo ich uzyskanie jest bardzo kosztowne lub mało wiarygodne
- Środowiska, gdzie system musi się dynamicznie dostosowywać do zmieniających się warunków (np. systemy sterowania)
- Rozwój zastosowań praktycznych, w których dostępne są masywne dane (Web, sensory, ...)
- ...
- **NIE ZAWSZE** warto stosować systemy uczące się!



# Systemy uczące się

---

**Uczenie** – zmiany w systemie adaptującym się pozwalające mu w przyszłości działać bardziej efektywnie na takich samych zadaniach lub zadaniach o podobnych charakterze (Simon 1983)



Nagroda Nobla w  
Ekonomii 1978

# Inne definicije

## Definition (#1/2)

A learning system uses  
sample data

to generate an updated basis  
for improved [performance]

on subsequent data  
from the same source

and expresses the new basis  
in intelligible symbolic form

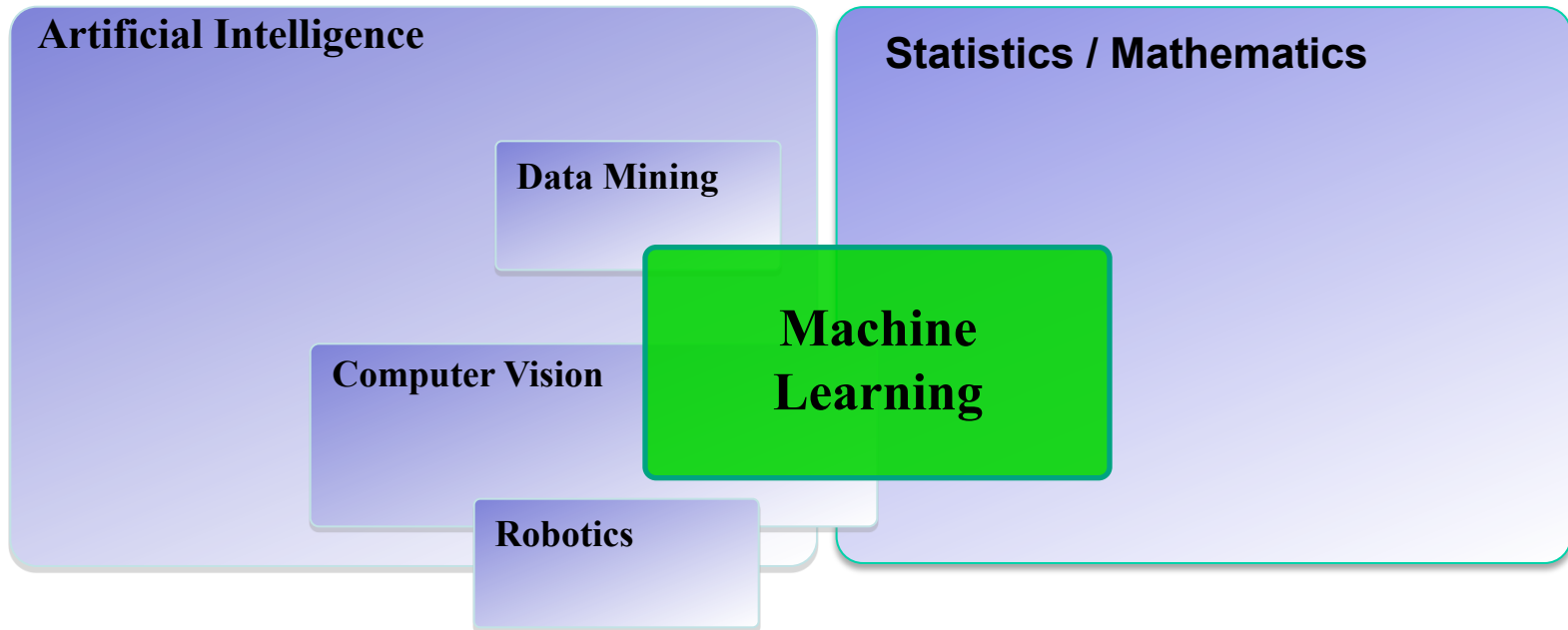


- Donald Michie (1991)

- *Computer Journal*

# *Machine Learning - powiązanie dziedzin*

---



(No definition of a field is perfect – the diagram above is just one interpretation, -)

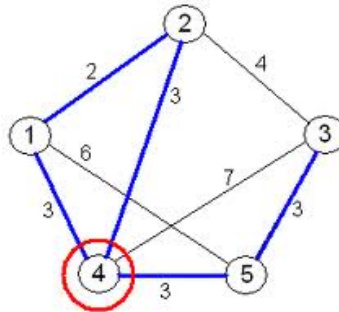
R. Mooney

# ML vs. Data mining (eksploracja danych)

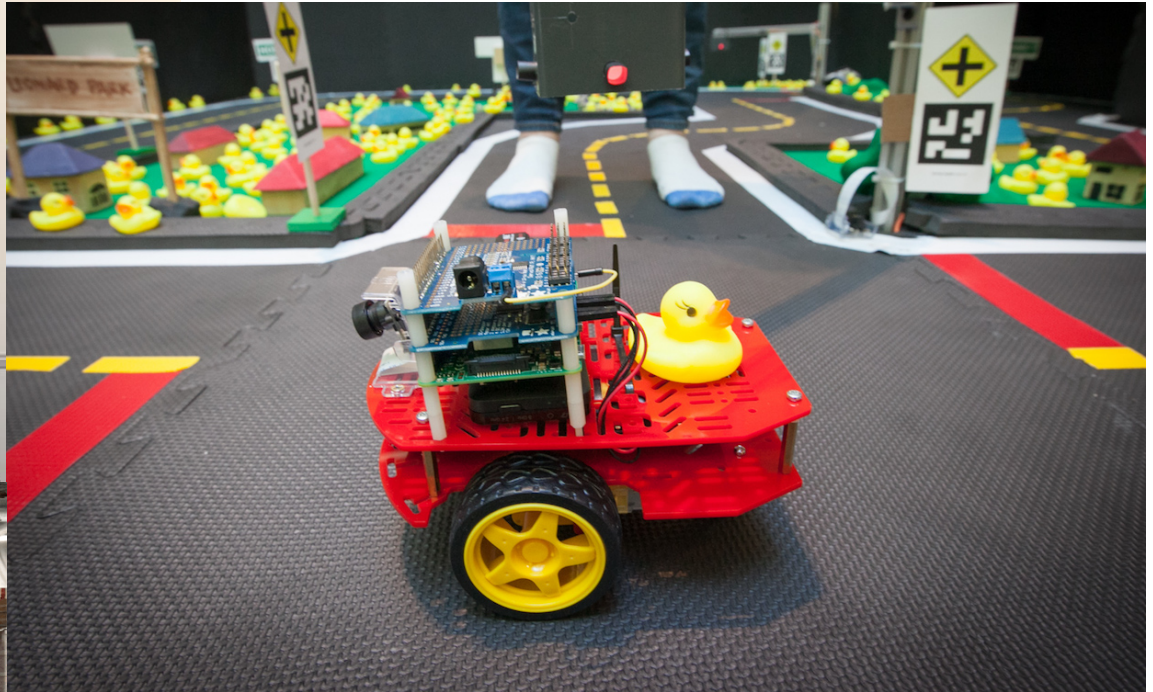
- **Motywacja** → zalew danych
  - Wzrost ilości danych generowanych przez nowe źródła pomiarowe
  - Potencjalnie duże / złożone repozytoria danych
- **Eksploracja danych** → Data mining
  - Poszukiwanie w zgromadzonych danych nieznanymi, użytecznych regularności, związków między elementami danych.
- Odkrywanie reprezentacji wiedzy (Knowledge Discovery from Data)



	A	B	C	D	E	F	G
1	AHRO Prevention Quality Indicators						
2	Dehydration Admission Rate (FGL 10)						
3							
4	Counties Marked in GREEN are significantly lower than the National Average.						
5	Counties Marked in RED are significantly higher than the National Average.						
6							
7	County Name	Cases	Population	Crude Rate	LCL	Risk Adj. Rate	Risk Adj. UCL
8	Adair	79	13,774	5.74	4.62	5.19	5.76
9	Allen	28	14,299	1.96	1.41	2.09	2.59
10	Anderson	12	15,453	0.78	0.25	0.84	1.42
11	Ballard	6	6,538	0.92	0.24	1.03	1.83
12	Barren	102	31,112	3.28	2.56	2.93	3.31
13	Bath	15	8,943	1.68	0.84	1.55	2.36
14	Bell	122	23,855	5.29	4.52	4.96	5.41
15	Benton	69	75,321	0.92	0.85	1.14	1.42
16	Bourbon	20	15,245	1.31	0.70	1.26	1.81
17	Boyd	32	39,393	0.81	0.39	0.72	1.08
18	Boyle	32	22,387	1.43	0.88	1.34	1.79
19	Breckin	18	6,700	2.69	1.78	2.63	3.47
20	Breathitt	40	12,361	3.23	2.84	3.59	4.16
21	Breckinridge	23	15,006	1.53	0.94	1.50	2.07
22	Bullitt	23	52,112	0.44	0.23	0.58	0.93
23	Butler	9	10,366	0.87	0.18	0.85	1.54
24	Caldwell	13	10,261	1.26	0.39	1.00	1.61
25	Callaway	26	29,185	0.95	0.50	0.93	1.30
26	Campbell	54	66,477	0.81	0.53	0.89	1.07
27	Carlisle	5	4,215	1.19	0.00	0.93	1.89
28	Carrill	20	7,950	2.52	1.77	2.56	3.35
29	Carter	18	21,160	0.85	0.37	0.85	1.34
30	Cass	47	12,648	3.72	2.72	3.39	3.89



# ML i Robotyka



Rola interakcji ze środowiskiem – sensory, akcje + wizja komputerowa / wykład prof. Skrzypczyńskiego PP-RAI

<https://pp-rai.cs.put.poznan.pl>

# Podstawowe typy uczenia się

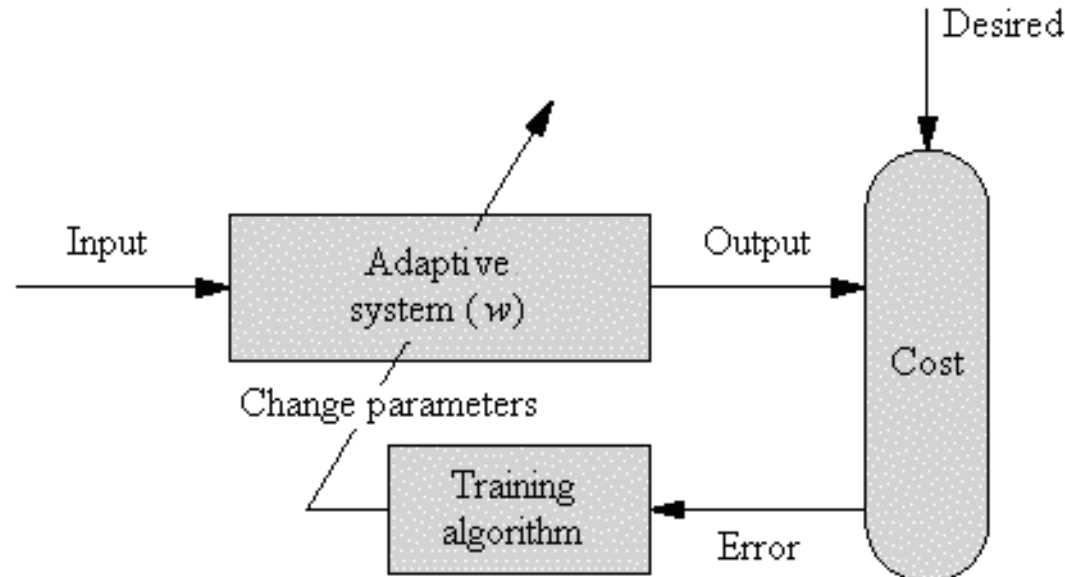
---

- **Supervised (inductive) learning**
  - Przykłady uczące (opis + wartość celu / etykieta – wyjście systemu)
- **Unsupervised learning**
  - Przykłady uczące (tylko opis bez pożądanego wyjścia)
- **Semi-supervised learning**
  - Dane częściowo etykietowane
- **Reinforcement learning**
  - Ocena (nagroda) jako rezultat sekwencji akcji

# Supervised learning / uczenie z nadzorem

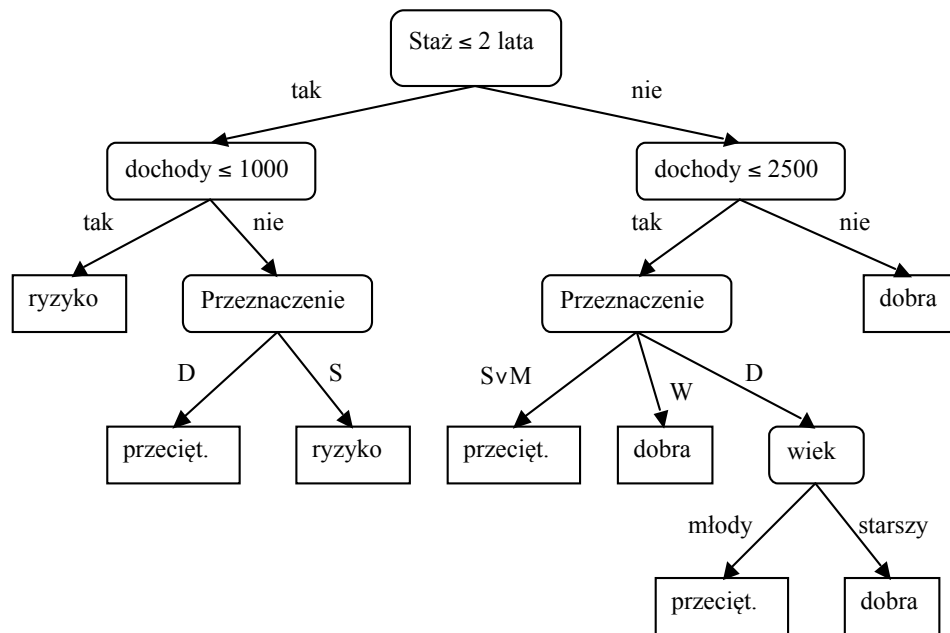
---

- Przykład uczący – opis cech charakterystycznych ( $\mathbf{x}$ ) i wymaganego wyjścia (target)  $y \rightarrow (\mathbf{x}, y)$ 
  - Dyskretne etykiety  $\rightarrow$  Klasyfikacja (ang.classification)
  - Wyjście liczbowe  $\rightarrow$  Regression, Numerical prediction



# Automatyczna indukcja drzew z przykładów

Lp.	Wiek	Sytuacja zawodowa	Staż pracy	Dochody	Cel kredytu	Ocena klienta
1	młody	uczy się	0	500	K	ryzyko
2	średni	Pracuje	2	1400	S	ryzyko
3	młody	pracuje	4	2600	M	dobra
4	starszy	pracuje	16	2300	D	dobra
5	średni	pracuje	14	1600	M	przecietna
6	młody	uczy się	0	700	W	ryzyko
7	średni	bezrobotny	0	600	D	ryzyko
8	młody	pracuje	3	1400	D	przecietna
9	średni	pracuje	11	1600	W	dobra
10	starszy	emeryt	0	1100	D	przecietna
11	młody	uczy się	0	1500	D	przecietna
12	młody	bezrobotny	0	1000	M	ryzyko
13	średni	pracuje	17	2500	S	przecietna
14	młody	bezrobotny	0	700	D	ryzyko
15	starszy	pracuje	21	5000	S	dobra
16	młody	pracuje	5	3700	M	dobra
17	młody	bezrobotny	0	800	K	ryzyko



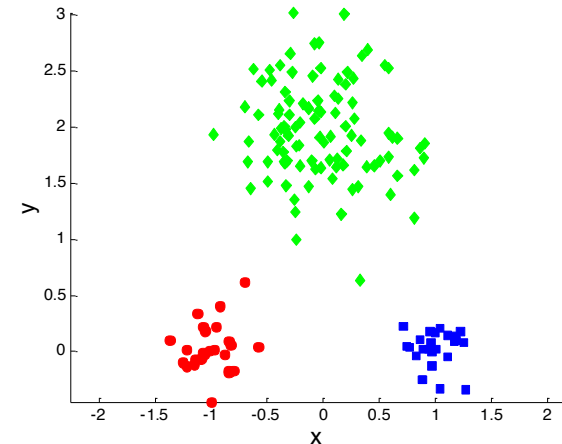
Algorytmy: C4.5 / ID3 lub CART



# Unsupervised Learning / uczenie nienadzorowane

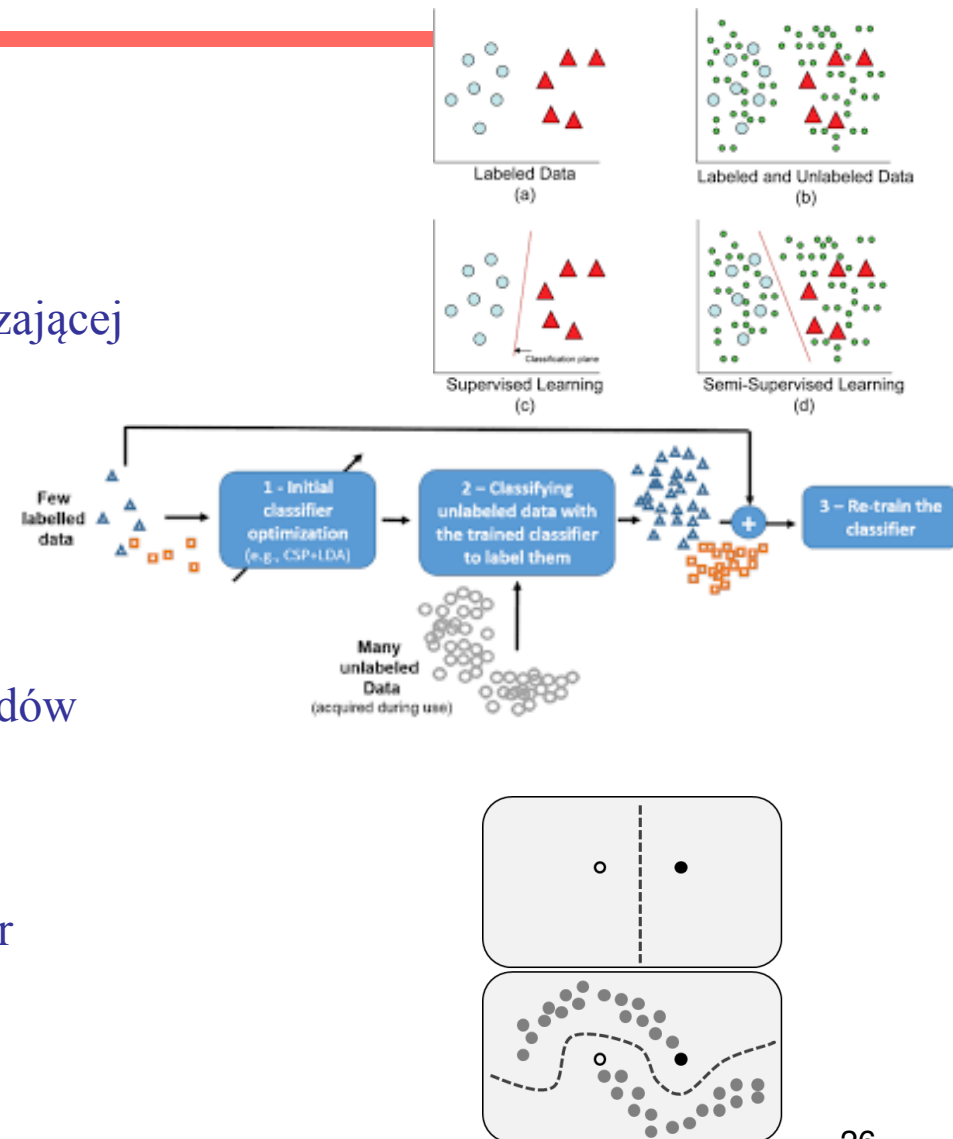
---

- Brak zmiennej wyjściowej / target label  $y$
- **Grupowanie**: poszukiwanie skupisk przykładów
  - Statystyka vs. Kohonen ANN
- Inne zastosowania w data mining: Data summarization, Association Analysis
- Problemy praktyczne
  - Customer segmentation in CRM
  - Kwantyzacja w przetwarzaniu obrazów
  - Kompresja danych
  - Wspomaganie systemów wyszukiwania informacji



# Uczenie częściowo nadzorowane

- Nieetykietowane przykłady
  - Łatwo dostępne i „tanie”
- Etykietowanie przykładów
  - Trudniejsze do pozyskania w wystarczającej ilości
  - Ludzki czynnik – drogi, zmęczony,
  - Rzadko automatyczny
- Typowa sytuacja
  - Ograniczona liczba w pełni etykietowanych przykładów
  - Duża pula nieetykietowanych przykładów
- Jak to wykorzystać do ulepszenia klasyfikatora wstępnie nauczonego na etykietowanych przykładach
  - Active Learning, Co-training, Transfer learning
- Jakie przykłady wybrać do zapytania wyroczni?



# Uczenie się ze wzmocnieniem

## - Reinforcement learning

---

- Dynamiczne interakcje ucznia ze środowiskiem, w którym działa, realizując swoje zadanie.
- Interakcje te odbywają się dyskretnych (na ogół) krokach czasu i polegają na obserwowaniu przez ucznia kolejnych stanów środowiska oraz wykonywaniu wybranych zgodnie z jego obecną strategią decyzyjną akcji.
- Po wykonaniu akcji uczeń otrzymuje rzeczywistoliczbowe wartości wzmocnienia lub nagrody, które stanowią pewną miarę oceny jakości jego działania.
- Wykonanie akcji może również powodować zmianę stanu środowiska.



# Klasyfikacja metod maszynowego uczenia się

---

- W ogólności system zmieniać się może w dwojaki sposób:
  - system nabywa "nową wiedzę" z zewnętrznych źródeł,
  - system modyfikuje się samodzielnie poprzez intensywne wykorzystywanie posiadanej już wiedzy w sposób bardziej efektywny.
- **Wnioskowanie Indukcyjne** opiera się na zbiorze faktów (obserwacji) oraz ewentualnym zbiorze hipotez a priori dotyczących tych faktów, a w wyniku daje uogólnienie wyjaśniające te fakty.
- **Dedukcja** - wyprowadzanie wniosków, które są logiczną konsekwencją posiadanej informacji.

## *Dimensions: Constraints*

*(on selecting a learning algorithm)*

- o Task/objective
  - Learning task
  - Performance task
- o Availability of background knowledge
  - Encoded
  - Interactive
- o Availability of data
  - Incremental vs. batch
  - Passive vs. active
- o Characteristics of data
  - Static vs. drifting
  - Propositional or first-order

## *Learning Tasks*

Supervised learning (from examples)

- Knowledge-poor
- Knowledge-intensive
  - *Improve predictive accuracy*
  - *Improve explanatory capability*
  - *Improve efficiency (speed, space)*

Problem solving

- Learning operator application conditions
- Learning to solve additional problems

Unsupervised learning (clustering)

- No teacher available; unlabeled data

Scientific discovery

- Often involves aspects of both supervised & unsupervised learning

# Dodatkowe uwagi o klasyfikacji metod

# Definicja problemu uczenia

---

**Uczenie = doskonalenie działania dla pewnych zadań na podstawie doświadczenia**

- doskonalenie działania dla zadań  $T$
- z względu na miarę oceny  $P$
- oparte na doświadczeniu  $E$

**Przykład 1:** Gra w szachy

$T$ : nauczyć się rozgrywać końcówki szachowe

$P$ : % wygranych gier

$E$ : zapis rozgrywek bardzo dobrych graczy

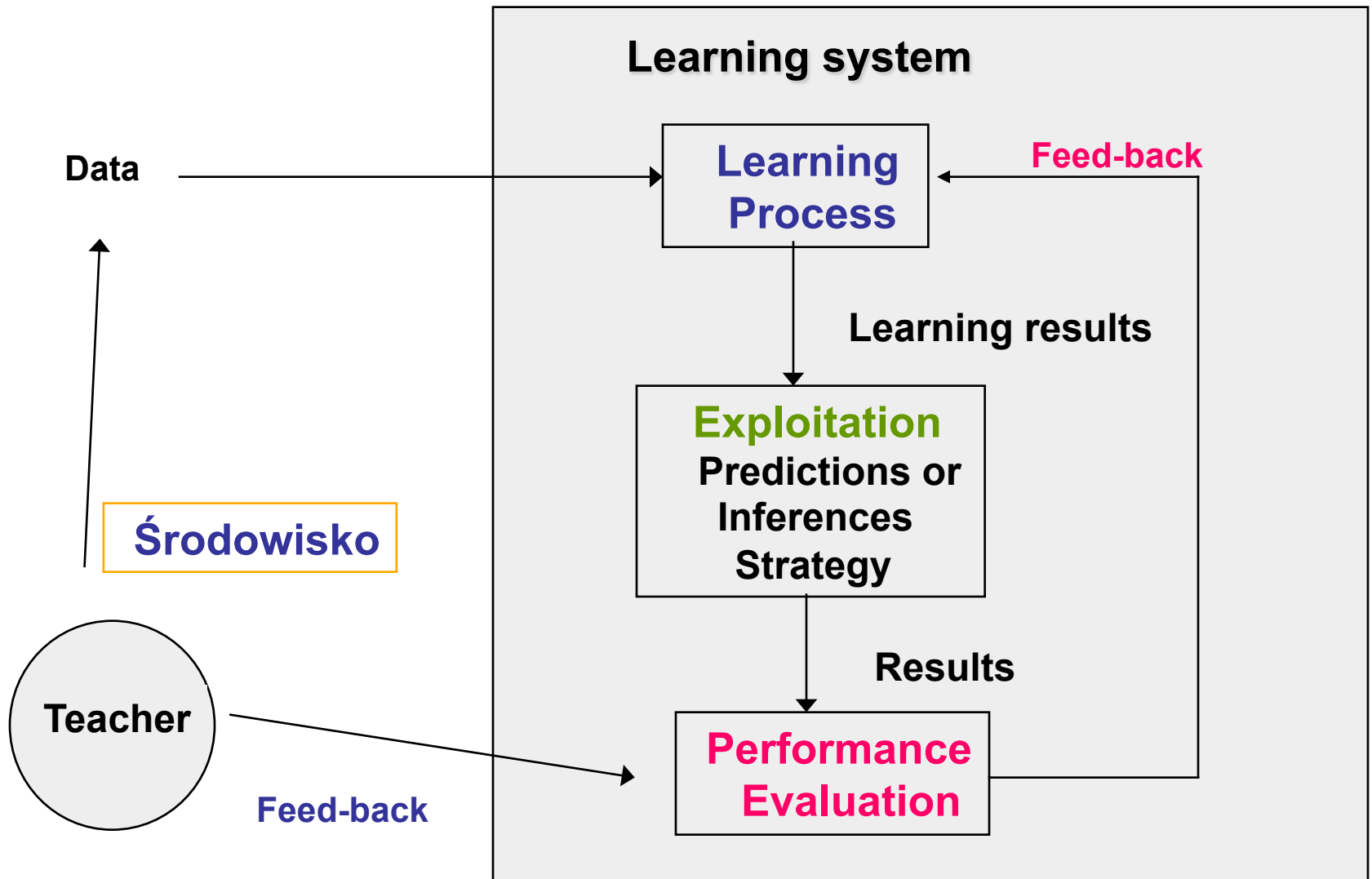
**Przykład 2:** Rozpoznawanie adresu (kodu) pocztowego

$T$ : rozpoznanie adresu (kodu) odbiorcy na liście

$P$ : % sortowanych listów poprawnie przypisanych do miasta odbiorcy

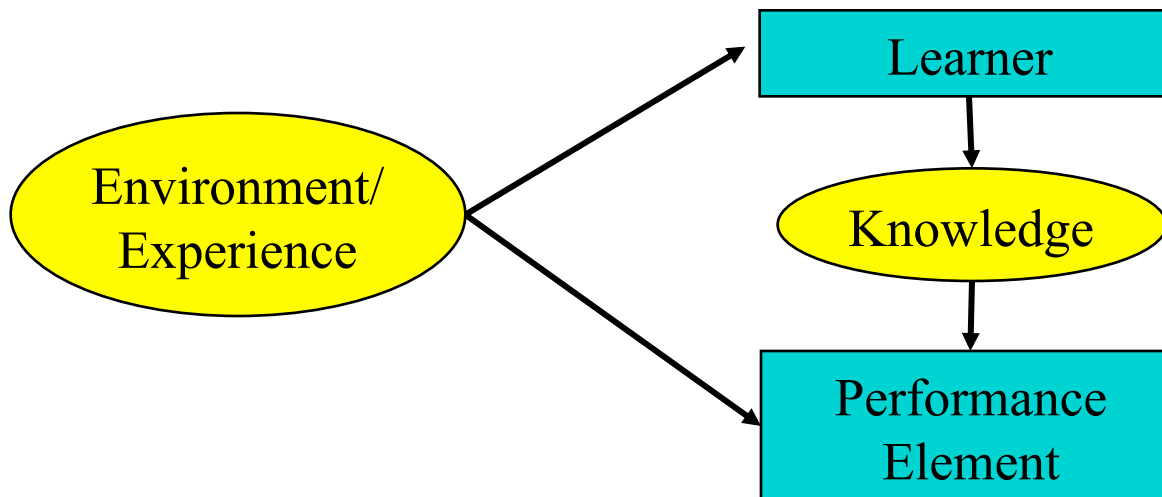
$E$ : odpowiednia liczba przykładów ręcznie napisanych adresów ze znanym adresem przeznaczenia

# Części procesu uczenia systemów



# Realizacja procesu uczenia się

- Jednoznaczne zadanie uczenia się
- Wybrane miary oceny (jedno, dobrze zdefiniowane kryterium).
- Co jest doświadczeniem (przykładami)?
- Czego należy się nauczyć?
- Jak reprezentować przykłady uczące?
- Jak reprezentować hipotezy/wiedzę?
- Jaki algorytm uczenia należy wybrać ?





# Różne reprezentacje



## Jawna reprezentacje wiedzy

- Numerical functions
  - Linear regression
- Symbolic functions
  - Decision trees
  - Rules in propositional logic
  - Rules in first-order predicate logic
- Instance-based functions
  - Nearest-neighbor
  - Case-based
- Probabilistic Graphical Models
  - Naïve Bayes
  - Bayesian networks
  - Hidden-Markov Models (HMMs)

## Brak jawnej reprezentacji (black box)

- Statystyczne złożone
  - Support vector machines
- Sieci neuronowe
- Zespoły klasyfikatorów
  - Bagging, boosting
  - ECOC, inne
- Złożone i hybrydowe systemy
  - Fuzzy Hybrid
  - Algorytmy ewolucyjne i inspirowane nature

## Inne kategorie

- Programowanie genetyczne
- Regresja symboliczna

# Optymalizacja / przeszukiwanie ...

---

## Optymalizacja ciągła

- Gradient descent
  - Model liniowy
  - ANN, Backpropagation

## Przeszukiwanie heurystyczne

- Divide and Conquer
  - Decision tree induction
  - Rule learning

## Evolutionary Computation

- Genetic Algorithms (GAs)
- Genetic Programming (GP)
- Nature inspired

## Optymalizacja z ograniczeniami

- Programowanie matematyczne
  - SVM

## Inne

- Dynamic Programming
  - HMM Learning
  - PCFG Learning
  - ...

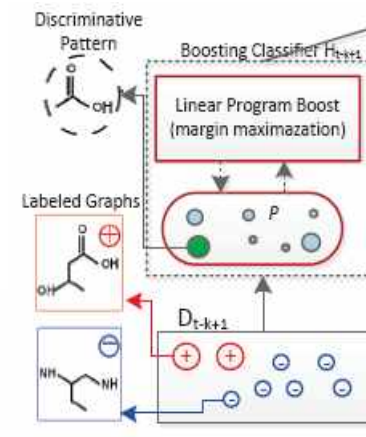
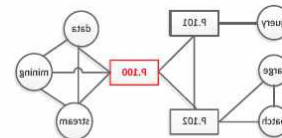
# Reprezentacja przykładów uczących – (atrybut wartość)

## Najprostsza reprezentacja tabelaryczna

- Przykład uczący  $x_i$  opisany wektorem  $\langle v_{i1}, v_{i2}, \dots, v_{ik} \rangle$  wartości atrybutów z  $A$  i etykietą klasy (target value  $y$ )

	$a_1$	...	$a_k$	Y / Classes
$x_1$	$V_{1,1}$	...	$V_{1,k}$	$V_{1,k+1}$
...	...	...	...	...
$x_i$	$V_{i,1}$	...	$V_{i,k}$	$V_{i,k+1}$
...	...	...	...	...
$x_n$	$V_{n,1}$	...	$V_{n,k}$	$V_{n,k+1}$

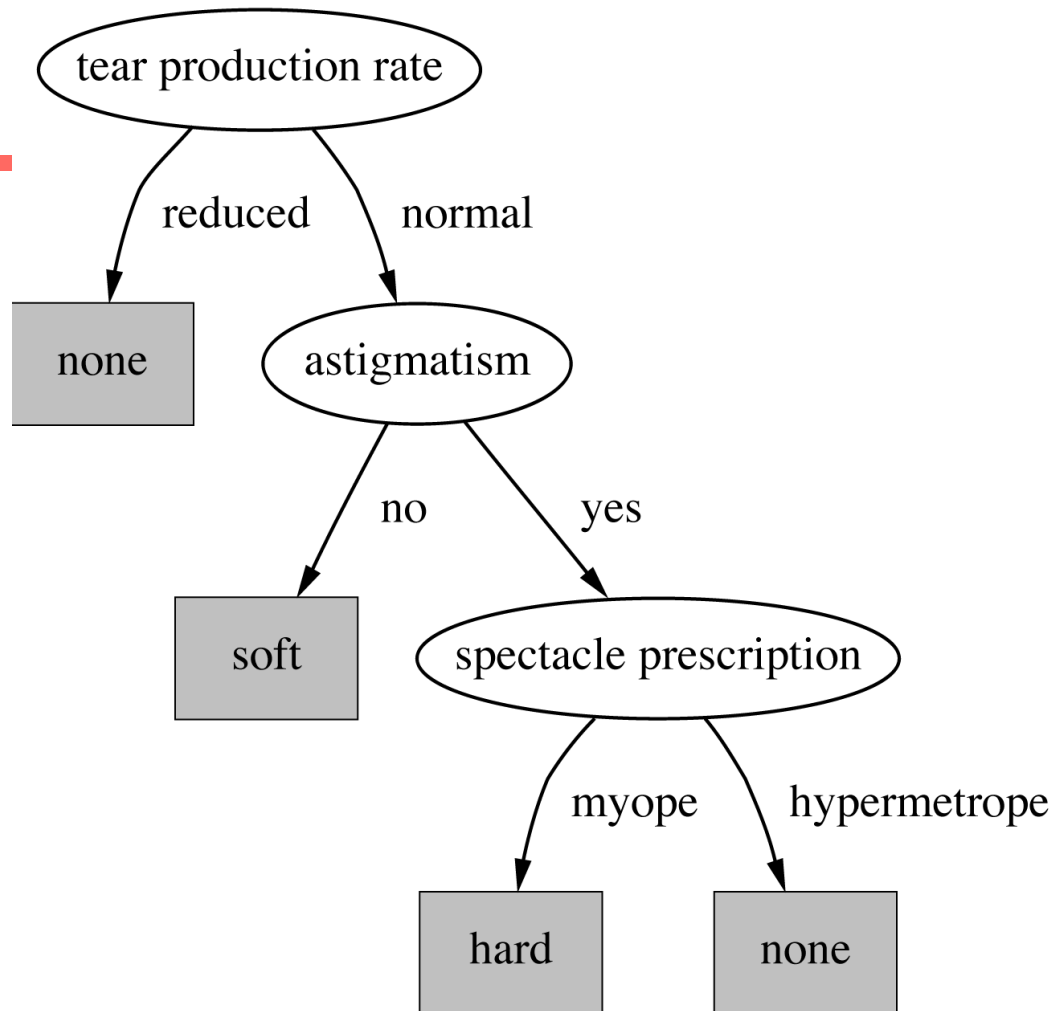
- Inne b. złożone reprezentacje – wielo-relacyjne zależności (ILP), grafy, obrazy lub sygnały, sekwencje dyskretne, połączenie wielu form podstawowych
- Big Data – efektywne obliczeniowo do dostępu oraz przetwarzania



# Reprezentacja przykład – tablica danych

- Tabela (U,AU{d})
- Problem tzw. contact lenses / dobór szkieł kontaktowych:
- Atrybuty:
  - age {young, pre-presbyopic, presbyopic}
  - spectacle-prescrip {myope, hypermetrope}
  - astigmatism {no, yes}
  - tear-prod-rate {reduced, normal}
- Decyzja contact-lenses {soft, hard, none}

age	specpres	astig	tearprod	contlen
young	myope	no	reduced	none
young	myope	no	normal	soft
young	myope	yes	reduced	none
young	myope	yes	normal	hard
young	hypermetrope	no	reduced	none
young	hypermetrope	no	normal	soft
young	hypermetrope	yes	reduced	none
young	hypermetrope	yes	normal	hard
pre-presbyopic	myope	no	reduced	none
pre-presbyopic	myope	no	normal	soft
pre-presbyopic	myope	yes	reduced	none
pre-presbyopic	myope	yes	normal	hard
pre-presbyopic	hypermetrope	no	reduced	none
pre-presbyopic	hypermetrope	no	normal	soft
pre-presbyopic	hypermetrope	yes	reduced	none
pre-presbyopic	hypermetrope	yes	normal	none
presbyopic	myope	no	reduced	none
presbyopic	myope	no	normal	none
presbyopic	myope	yes	reduced	none
presbyopic	myope	yes	normal	hard
presbyopic	hypermetrope	no	reduced	none
presbyopic	hypermetrope	no	normal	soft
presbyopic	hypermetrope	yes	reduced	none
presbyopic	hypermetrope	yes	normal	none



**Decision tree** for the contact lens data  
Drzewo decyzyjne – drzewo klasyfikacyjne

# Terminologia związana z indukcyjnym uczeniem się

---

- Dziedzina  $X$
- Przykład uczący  $x \in X$
- Atrybut  $a_i$
- Klasa, pojęcie docelowe  $C \subseteq X \quad c: X \rightarrow \{0,1\}$  (0-1 pojęcia)
- $C_i = \{x \in X : c(x)=i\}$
- Przykład etykietowany dla uczenia z nadzorem  $\langle x, c(x) \rangle$
- Zbiór uczący – wiele przykładów
- Przykład pozytywny dla pojęcia  $c$  - taki  $x \in X$  że  $c(x)=1$
- Przykład negatywny dla pojęcia  $c$  - taki  $x \in X$  że  $c(x)=0$
- Hipoteza  $h$  - funkcja  $f: X \rightarrow \{0,1\}$
- Przestrzeń hipotez  $H$
- Błąd stosowania hipotezy  $h$  dla zbioru treningowego  $D$
- Indukcyjne ukierunkowanie (ang. *inductive bias*)

# Uczenie się pojęć

---

- **Pojęcie** (ang. *concept*) – ogólne określenie oznaczające zbiór obiektów posiadających wspólne właściwości, które odróżniają je od innych pojęć.
- Pojęcie to jedna z postaci wiedzy o świecie, używana do opisywania systemów interpretowania obserwacji oraz abstrakcyjnych idei.
- Pojęcia – pozwalają na klasyfikowanie pewnych obiektów na zbiory (kategorie / klasy)  $x: c(x) \in C$ .
- Pojedyncze pojęcie wyznacza podział zbioru obiektów na **przykłady pozytywne** oraz **przykłady negatywne** (kontrprzykłady) tego pojęcia  $C = \{0, 1\}$  ( $\{\text{true}, \text{false}\}$ ).
- Pojęcia wielokrotne – klasyfikacja  $C = \{c_1, c_2, \dots, c_n\}$

# Formalizacja

---

- Przykład uczący  $x \in X$ , etykietowany, czyli  $\langle x, c(x) \rangle$  dla pewnej nieznanej funkcji klasyfikującej  $c$ .
- Dany zbiór przykładów uczących  $D$ 
  - Wybór pewnego pojęcia – przykłady pozytywne i negatywne tego pojęcia
- Algorytm ma automatycznie odkryć pewną hipotezę  $h(x)$ , taką, że:

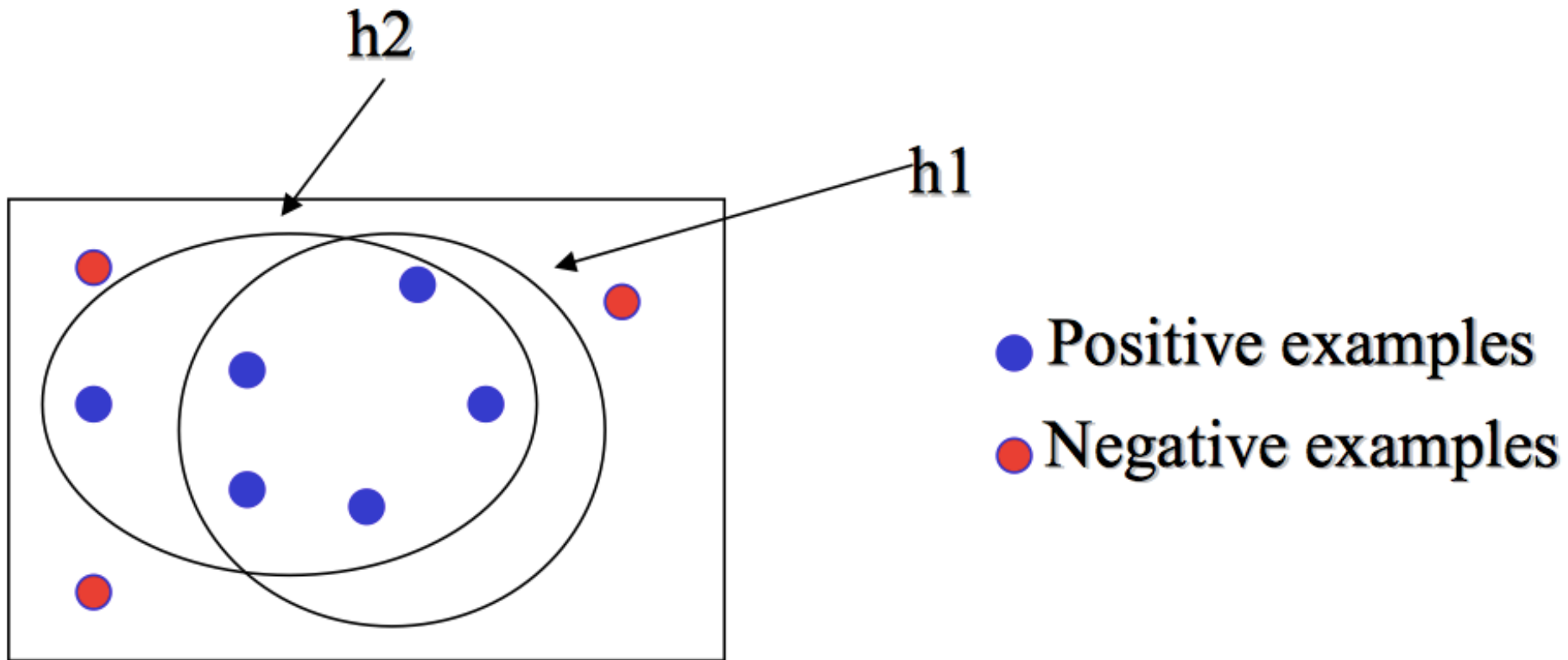
$$\forall \langle x, c(x) \rangle \in D : h(x) = c(x)$$

**Wymóg - spójności**



# Różne możliwe spójne hipotezy

---



Spójność hipotez vs. pokrywanie przykładów

# Przykładowy problem

---

- Instance language:  $\langle \text{size, color, shape} \rangle$ 
  - $\text{size} \in \{\text{small, medium, large}\}$
  - $\text{color} \in \{\text{red, blue, green}\}$
  - $\text{shape} \in \{\text{square, circle, triangle}\}$
- $C = \{\text{positive, negative}\}$

- $D$ :

Example	Size	Color	Shape	Category
1	small	red	circle	positive
2	large	red	circle	positive
3	small	red	triangle	negative
4	large	blue	circle	negative

# Reprezentacja hipotez

---

- Reprezentacja koniunkcyjna lub DNF;
- Można zbudować wiele możliwych hipotez
  - red & circle
  - Large & red
  - (small & circle) or (large & red)
  - (small & red & circle) or (large & red & circle)
  - not [ ( red & triangle) or (blue & circle) ]
  - not [ ( small & red & triangle) or (large & blue & circle) ]
- Bias (indukcyjne ukierunkowanie)
  - Any criteria other than consistency with the training data that is used to select a hypothesis among many possible.

# Generalizacja – uogólnienie dla predykcji

---

- Wyindukowana hipoteza z danych musi być także stosowalna dla przykładów spoza zbioru uczącego (generalizacja - uogólnienie / predykcja przyszłych faktów)
- Zapamiętanie wszystkich przykładów uczących – słaba zdolność uogólniania
- *Occam's razor*:
  - Finding a *simple* hypothesis helps ensure generalization.

# Conjunctive Rule Learning

- Conjunctive descriptions are easily learned by finding all commonalities shared by all positive examples.

Example	Size	Color	Shape	Category
1	small	red	circle	positive
2	large	red	circle	positive
3	small	red	triangle	negative
4	large	blue	circle	negative

Learned rule: **red & circle** → positive

IF (color=red) & (shape=circle) THEN (category=positive)

- Must check consistency with negative examples. If inconsistent, **no** conjunctive rule exists.

# Limitations of Conjunctive Rules

- If a concept does not have a single set of necessary and sufficient conditions, conjunctive learning fails.

Example	Size	Color	Shape	Category
1	small	red	circle	positive
2	large	red	circle	positive
3	small	red	triangle	negative
4	large	blue	circle	negative
5	medium	red	circle	negative

Learned rule: red & circle → positive

Inconsistent with negative example #5!

# Disjunctive Concepts

- Concept may be disjunctive.

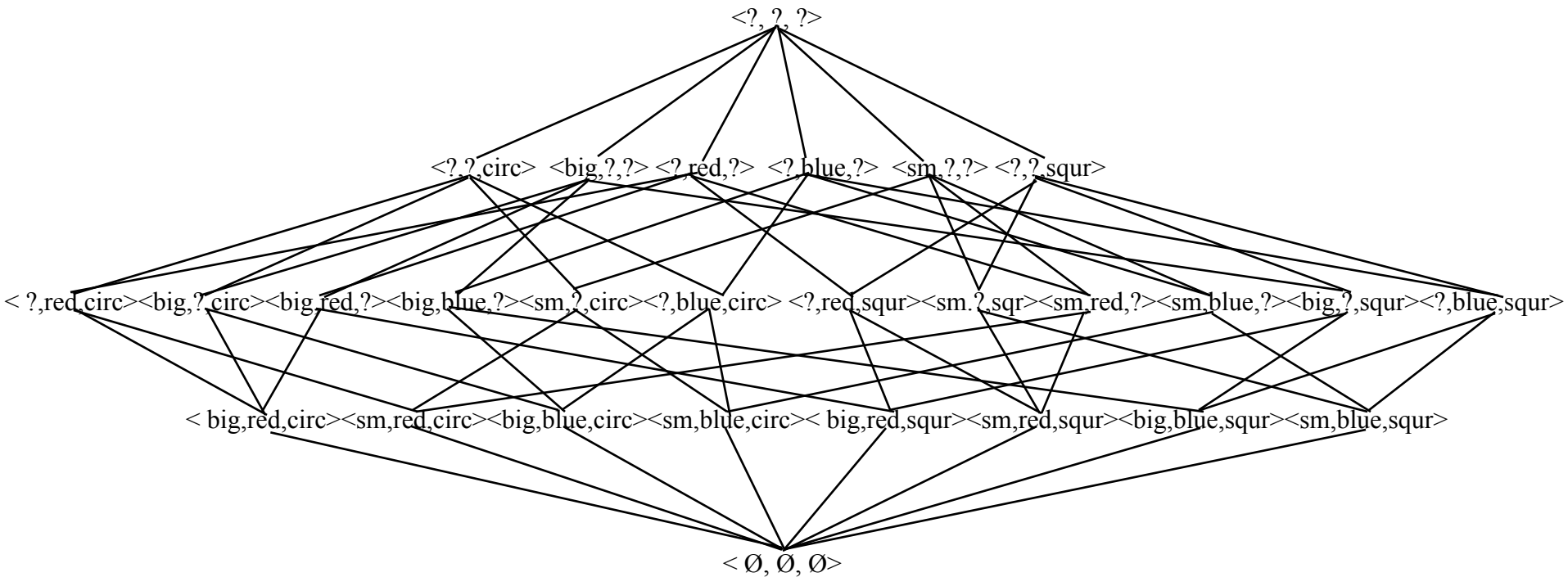
Example	Size	Color	Shape	Category
1	small	red	circle	positive
2	large	red	circle	positive
3	small	red	triangle	negative
4	large	blue	circle	negative
5	medium	red	circle	negative

Learned rules: small & circle → positive  
large & red → positive

# Przeszukiwanie przestrzeni hipotez

---

Size: {sm, big}    Color: {red, blue}    Shape: {circ, squar}



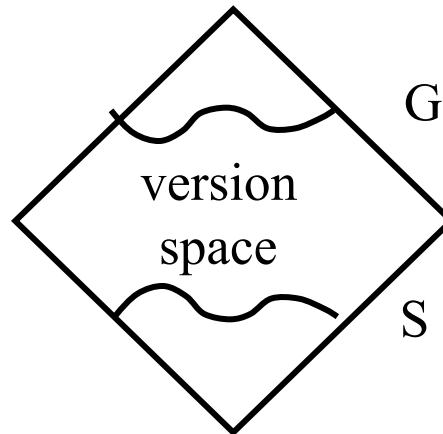
Number of hypotheses =  $3^3 + 1 = 28$



# Zadanie domowe

---

- Systematyczne podejście do uczenia się pojęć w rozważanych reprezentacjach:
- Przestrzeń wersji (CEA) – Algorytm eliminacji kandydatów (T.Mitchell)



# Elementy CEA na które warto zwrócić uwagę

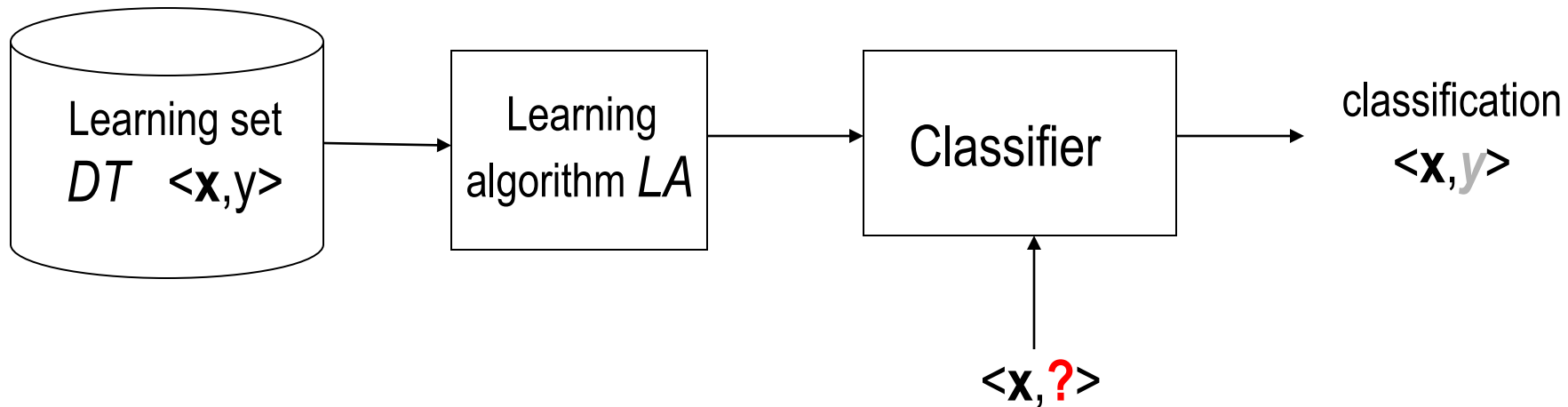
---

- Uporządkowanie przestrzeni hipotez
- Zbieżność algorytmów do właściwej poprawnej funkcji etykietującej / hipotezy
- Rola ograniczeń G i S
  - Operacje uogólnienia oraz uszczegółowiania hipotez
- Trudności z analizą błędnych / sprzecznych przykładów
- Klasyfikowanie nowych przykładów
  - (zwłaszcza, jeśli G i S nie zbiegły się)
- Wpływ kolejności analizy przykładów uczących
- Możliwości zadania pytań przez system – proponowanie przykładu do etykietyzacji (tak aby ograniczyć przeszukiwanie przestrzeni)
- Pojęcie „inductive bias” – język reprezentacji

# Ocena klasyfikatorów

Classification accuracy =  $n(\text{correct})/n_{\text{test}}$

Trafność klasyfikowania



		True class	
		p	n
Hypothesis output	Y	TP (True Positives)	FP (False Positives)
	N	FN (False Negatives)	TN (True Negatives)

Column counts:  $P_c$   $N_c$

Niezależny zbiór przykładów testowych

Inne miary

- Analiza macierzy pomyłek
- Deterministyczne (czułość, specyficzność) vs. metody graficzne (ROC – AUC)

Tzw. funkcje straty (ang. loss functions)

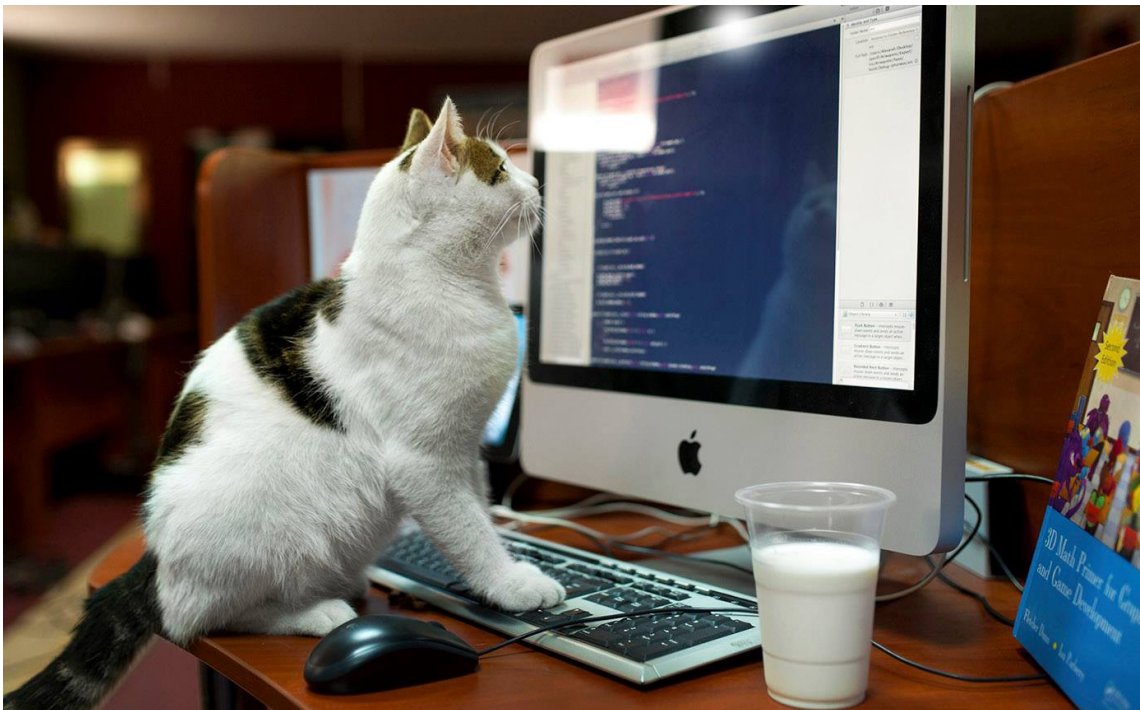
# Ocena systemów uczących się

---

- Experimentalna
  - Kontrolowana ocena krzyżowa (ang. cross-validation) algorytmów na zbiorze tzw. benchmark datasets
  - Decyzja o miarach oceny predykcja, wydajność, itd.
  - Statystyczna analiza różnic (zwłaszcza przy porównaniu algorytmów)
- Teoretyczna
  - Matematyczne podejście (twierdzenia, analiza formalna):
    - Ocena złożoności
    - PAC nauczalność (czy można się nauczyć danego pojęcia)
    - Sample complexity (liczba przykładów potrzebna do nauczenia się wybranej rodziny funkcji klasyfikującej)
    - VC - wymiar

# Próbuj samemu i szukaj więcej!

---



## Rozwiązuj sam problemy wymagające ML!

# Koniec wykładu – dziękuję za uwagę

---

## Pytania lub komentarze?

### Zapraszam na „konsultacje”



Czytaj książki, artykuły, ...

Kontakt:

[Jerzy.Stefanowski@cs.put.poznan.pl](mailto:Jerzy.Stefanowski@cs.put.poznan.pl)

<http://www.cs.put.poznan.pl/jstefanowski>



CAMIL Research Center