Ingineria Sistemelor de Inteligenta Artificiala

Corneliu Florea

B143, Leu

Corneliu.Florea@upb.ro

Administrativ

2 teste : saptamana 6- 15 %
saptamana 12 -15%
(discutam pe parcurs organizarea)

Predare proiect: saptamana 11 - **50 %**
predare mai tarziu (ianuarie-septembrie) 40%

Colocviu - 30%
(NU se reface)

Proiect

Implementare functionala pentru o baza de date:

- Baza de date este din “UCI repository”:
- https://archive.ics.uci.edu/ml/datasets.html
- Baza de date este nominala - liste de studenti:
- **1 sistem antrenabil bazat pe orice librarie vreti**
  - **Sistemul** este dintre: masina cu vectori suport (SVM), random forest (RF), Adaboost peste arbori, Retea neurala
    - **Librarii: Python - scikit-learn (laborator - recomandat), Matlab, Weka, OpenCV, etc.**
- **Raport** (1-3 pag, printat sau scris de mana)**:**
  - Ce problema rezolvam, In ce consta baza de data (cate exemple, cate dimensiuni)
  - Ce librarie am folosit
  - Impartit intre “train” si “test” - fie provine din baza de date, fie 75-25
  - Rezultate
    - ❑ Ce metrica
    - ❑ Variatia parametrilor - specifica fiecarui sistem

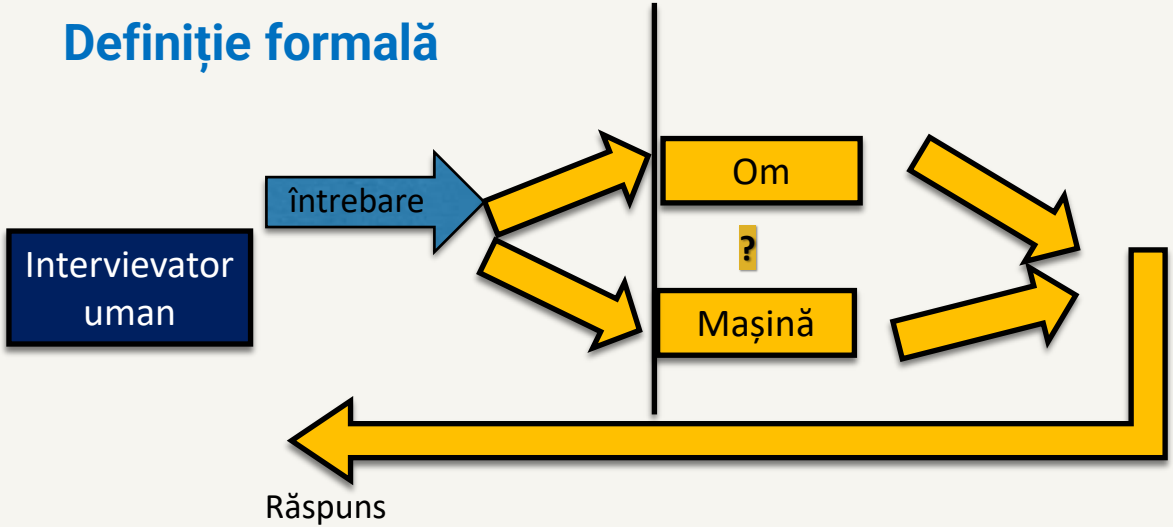
Proiect- sistemele

- **Masina cu Vectori Suport :**
  - nucleu liniar.
  - se variaza “Cost”: 2^{-5}, 2^{-3}, ... 2^{7}
- **Random Forest cu 10 arbori**
  - Se variaza concomitent (toate combinatiile posibile)
    - procentul in-bag - 25%,50%, 85%
    - Numarul de dimensiuni alese intr-un nod 10%, 50% 80%
- **Retea neurala** (Perceptron Multi-Strat)
  - Se variaza concomitent (toate combinatiile posibile)
    - Numarul de straturi ascunse - 1 sau 2
    - Numarul de neuroni pe straturile ascunse: egal cu stratul anterior sau jumate
    - Learning rate: 0.1 sau 0.01

INTRODUCDERE - PROBLEMATICA

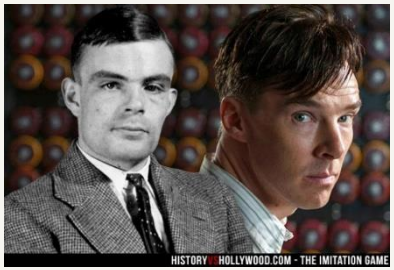
Testul Turing

Definiție formală



Daca intervievatorul nu poate sa isi dea seama daca a raspuns un om sau o masina atunci sistemul este un **agent inteligent!!**

Alan Turing



Imitation game

Turing (1950) "Computing machinery and intelligence"

Verificare 1

Care dintre următoarele sunt agenți inteligenți

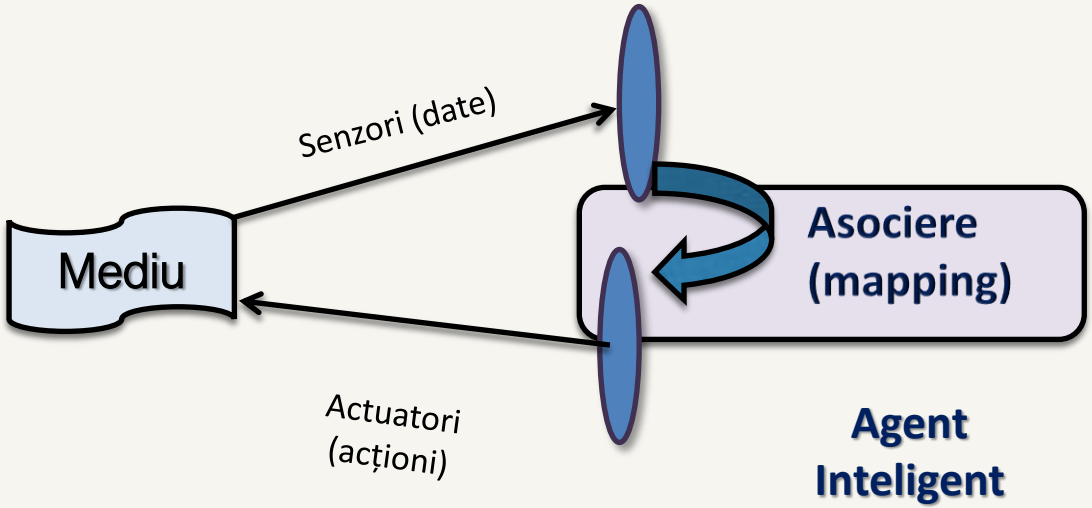
(a) Masina de scris

(b) Avionul modern

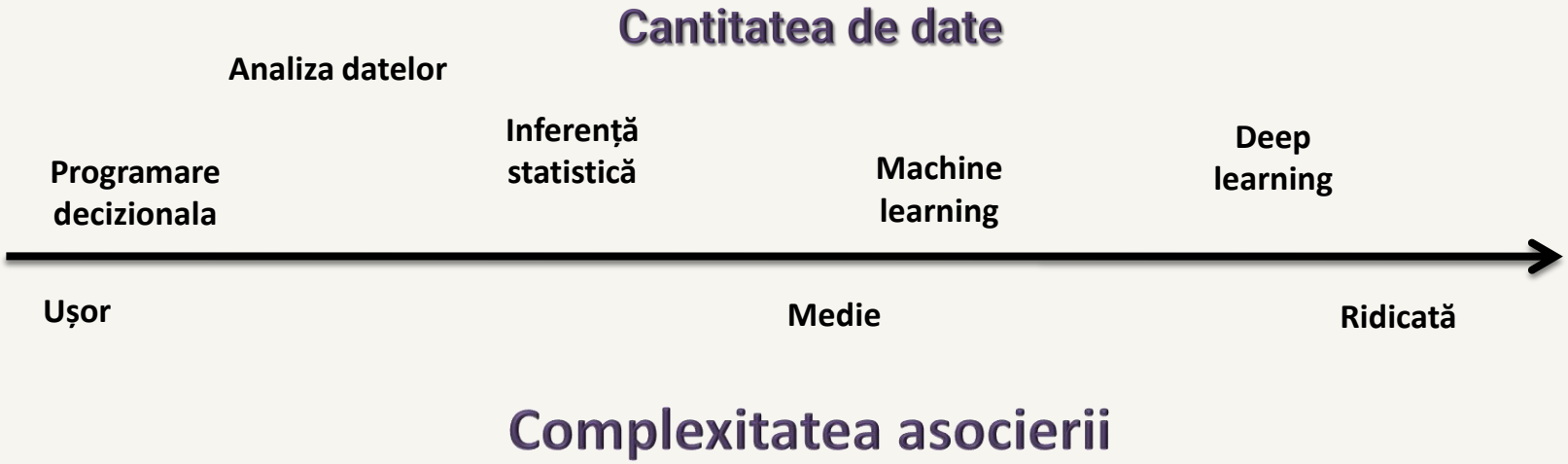
(c) Robotul chirurgical

(d) Program care controleaza dusmanul intr-un joc

Definiții


Senzori (date)
Mediu
Asociere
(mapping)
Actuatori
(acționi)
Agent
Inteligent

Definiții


Cantitatea de date
Analiza datelor
Inferență statistică
Programare decizionala
Machine learning
Deep learning
Ușor
Medie
Ridicată
Complexitatea asocierii

Relația AI - ML

- **Inteligenta Artificiala (AI)**
  - Denumeste un produs care are o componenta autonoma
  - Un ansamblu de parti
  - Domeniul AI include ML
  - Denumire la modă in 1990- 2010
  - Incepe sa devina perimata
- **Machine learning (ML)**
  - Reprezinta tehnica de construit o componenta autonoma
  - Un mic creier care rezolvă o problemă
  - Denumire la modă acum

Cum privesc oamenii ML

•“A breakthrough in machine learning would be worth ten Microsofts” (Bill Gates, Chairman, Microsoft)

•“Machine learning is the next Internet” (Tony Tether, Director, DARPA)

•Machine learning is the hot new thing” (John Hennessy, President, Stanford)

•“Web rankings today are mostly a matter of machine learning” (PrabhakarRaghavan, Dir. Research, Yahoo)

•“Machine learning is going to result in a real revolution” (Greg Papadopoulos, CTO, Sun)

•“Machine learning is today’s discontinuity” (Jerry Yang, CEO, Yahoo)

[Pedro Domingos]

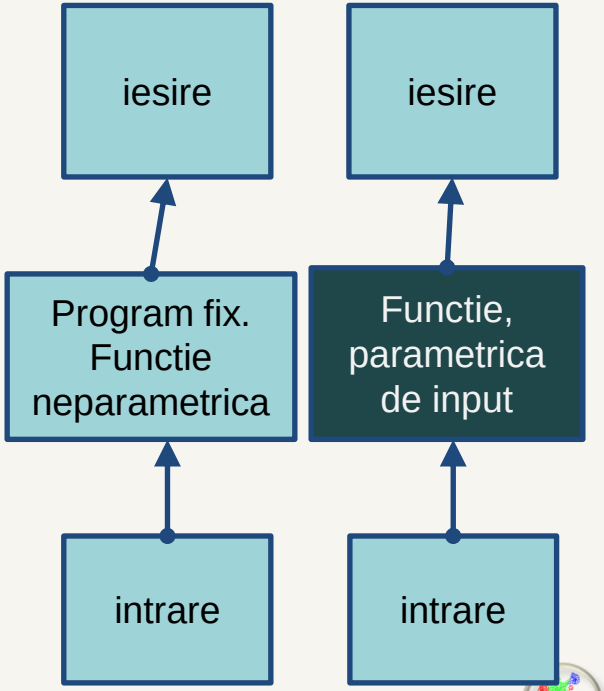
Ce încercam sa rezolvam?

- Lumea noastra este **plina de date**.
- Dupa ce le colectam si organizam, **datele**, cu putin noroc, se transformă în **informație**.
- Dificultatea este sa intelegem, integram datele si extragem informatia pentru a obtine **cunoștiințe utile**

- *Suntem inundați de date, dar vrem cunostiinte !!*

Abordari in AI

- **Abordare bazata pe reguli fixe** (Hand crafted rules)
  - Se codeaza (non parametric - Hard-code) informatia in limbaj formal
  - 1960 Nu foarte de succes **- prea multa munca**
- **Abordări de tip Machine Learning**
  - Informația este extrasa din date reale in mod automat
  - Pare foarte promitaoare in momentul de fata
  - Dar... ce fel de informatii... Cum putem învața o functie intrare-iesire



Sistem de ML : Definiții

- Sistem de ML = Funcție (parametrică) multidimensionala
- Trebuie sa faca legatura cat mai bine intre **intrare** si **iesirea** dorita
- **Antrenare** - cautam valorile parametrilor care produc cea mai buna legatura.
  - Problema de optimizare
- Date de antrenare – perechi intrare-iesire dorita

Cuprins curs

- Introducere: exemple; defintii; problematica
- Algoritmi de optimizare: problema; Newton; gradient
- Clasificare: Cel mai apropiat vecin. Clustering
- Arbori de decizie. Ansambluri de arbori
- Perceptron. Perceptron multi-strat
- Masini cu vectori suport

Relații cu alte materii

- Datele au o componentă aleatoare
  - Matematici speciale
  - DEPI
- Algoritmii (partea de antrenare dar si de test) se programeaza
  - Programarea calculatoare, SDA
  - POO
- Aprofundare
  - Machine Learning pentru Aplicatii Vizuale- master TAID

Exemplu

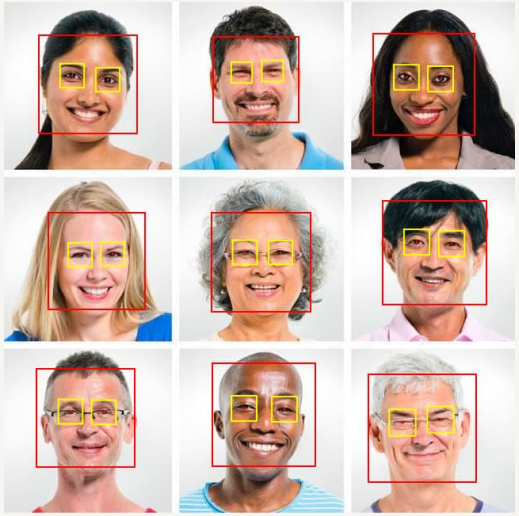
Caz simplu: *“X si 0”*

- Reguli clare, programabile
- Variabilitate redusa
- Strategia poate fi codata in clar

- Data?
  - Pozitii pentru X si respectiv 0 (vector de 9 valori)
  - Cine castiga: X, remiza sau 0
- Totul (toate datele) sunt utile?
  - Da

Exemplu

- Caz mediu – *Detectia fetelor intr-o imagine*
- Care sunt datele?
  - Pixelii (punctele din imagine)
  - Iesirea: e sau nu fata
- Toate datele sunt utile ?
  - Hmm!?
- Informatia – grupuri cu o anumita specificitate
- Cum o rezolvam?
  - La fiecare locatie ne intrebam daca grupul de pixeli ce ne inconjoara are forma de fata.



Exemplu

Relativ simple : *“Intensitatea luminoasa optimala pentru un smartphone”*

- Lumina de zi – contrast mare
- Night / dim light – contrast redus

Datele?

- Informatii de la toti cei 20 de senzori din telefon (camea, orientare…)
- Iesire ...

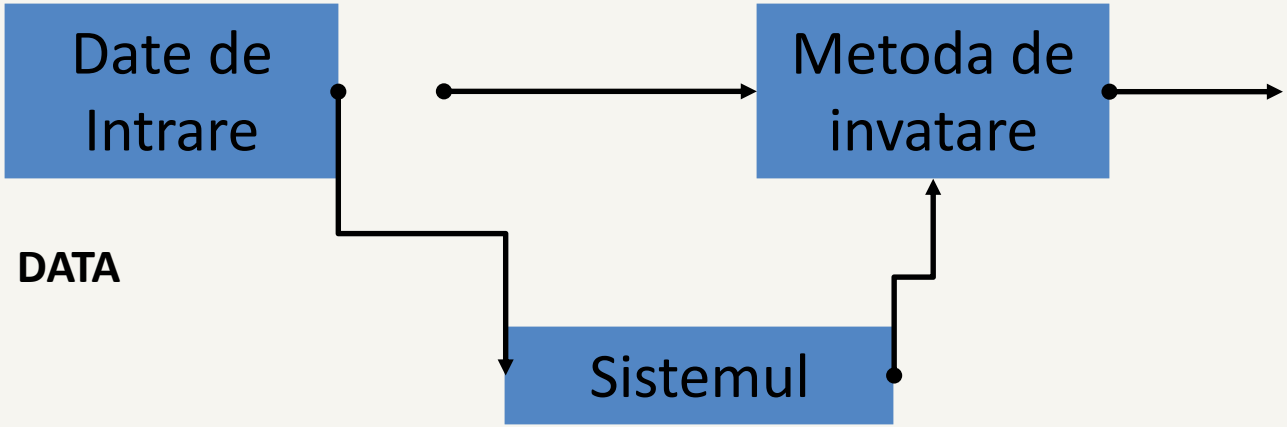
- Toate datele sunt utile?
  - De la accelerometri???
- Exista o solutie unica, unanim acceptata?

Exemplu

- Caz foarte dificil – *Masina autonoma*
- Data:
  - multe camere, radare montate pe masina (in interior si exterior)
  - Indicatii ale sistemelor masinii (viteza, senzori de frana, presiune pneuri, etc)
  - Date GPS, date despre trafic
- Ce e util ?
- Care e ieșirea dorită?
  - La fiecare moment de timp
  - Pe un task

Modelul unui sistem antrenabil

Testare: modelul antrenat prezice valori



Antrenare: cauta sa identifice parametri care permit modelului sa asocieze optim intrarea de iesire

Datele

- Datele – se structureaza
- Pentru cazul solutiei cu PC setul total de date este impartit intre:
  - Set de antrenare - se foloseste pentru a cauta valorile parametrilor necunoscuti in timpul procesului de antrenare
    - Set de Validare - Subset al setului de antrenare folosit pentru a valida performanta
  - Setul de testare – parte a datelor care nu e vizibila in timpul antrenarii
    - In mod normal setul de test e disponibil abia cand functioneaza on-line

Exemplu

- Recunoasterea fetelor (autentificare)
  - Un grup de 12 oameni lucrează într-un laborator
  - Avem 10 imagini cu fiecare fata in setul de antrenament
    - Setul de antrenare = 120 imagini
    - Fiecare imagine este asociata cu persoana din ea
  - Testare:
    - Fiind data o noua imagine, trebuie spus care dintre cele 12 persoane este in ea

Recunoasterea fetelor

Exemple de imagini din setul de antrenare

Mihai Viteazul

Imagini de test

Persoane:

1. Ileana Cos
2. Mihai Viteazul
3. Maria V.
4. Stefan Marescu
5. Soliman Gus
6. ...

Datele

- Sunt un vector multi-dimensional
  - Conține:
    - “vector de trăsături” – set of masuratori care descrie fiecare caz
    - Etichete:
      - Discrete – problema este de clasificare
        - » Binara: A sau B
        - » Categoriala (multi-class): A sau B sau C sau D….
      - Continua – regresie sau predictie

Predicție: Regresie

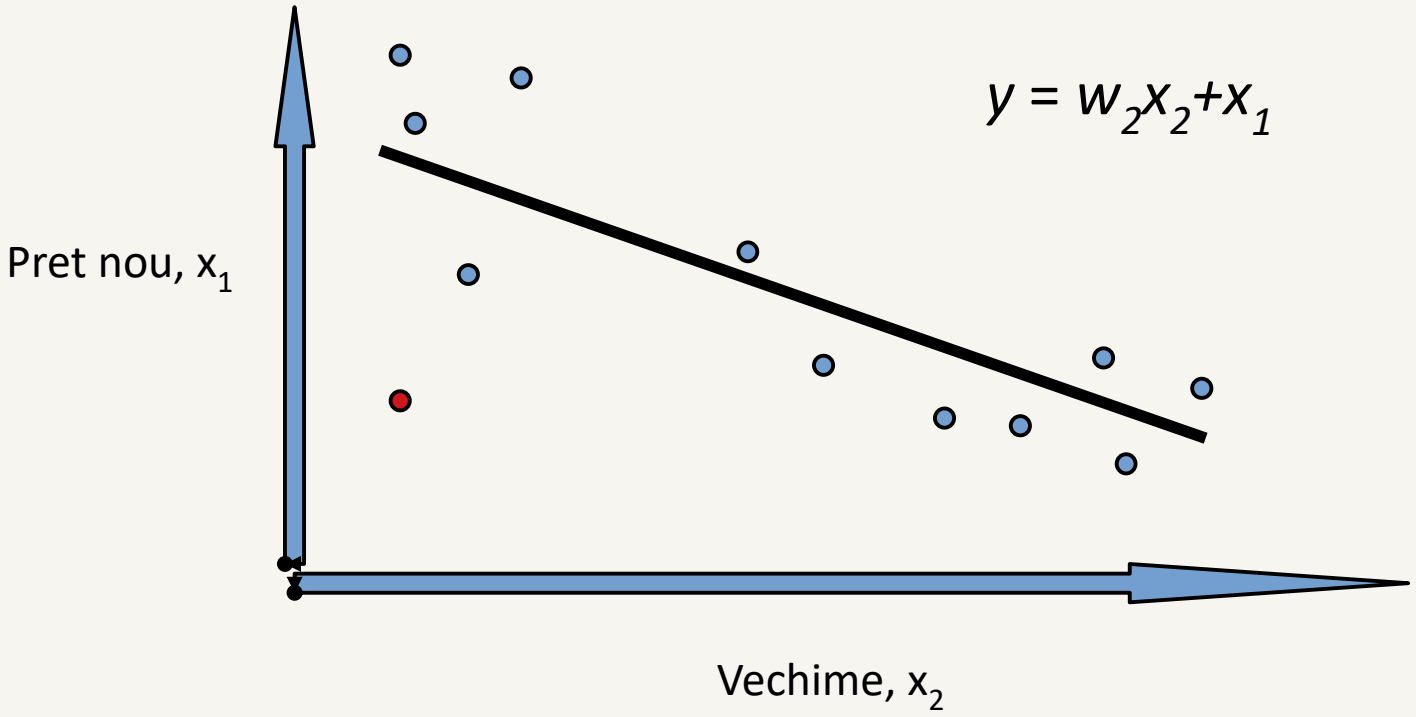
- Exemplu: Pretul unui apartament vechi
- x : atribute ale apartamentului
  - E.g. vechime
  - Valoare de nou

y : pretul curent

$$y = g(x \mid \theta)$$

$g(\ )$ modelul,

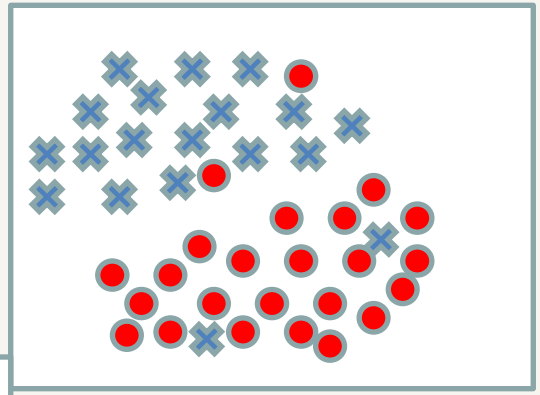
θ parametri modelului


$y = w_2x_2+x_1$
Pret nou, x_1
Vechime, x_2

Antrenare si testare

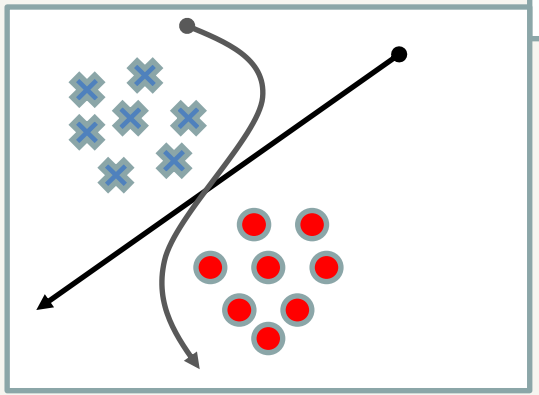
Problema de clasificare binara

Achizitia de date:
Esantionam setul real

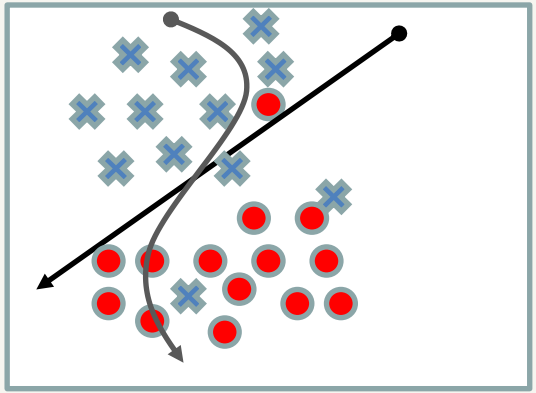


Setul de date adevarat si complet (**neobservat**)

In practica



Setul de antrenare
(observat)



Setul de testare
(ne-observat)

Sisteme antrenabile

Învatare supervizata
(toate data **au etichete**)

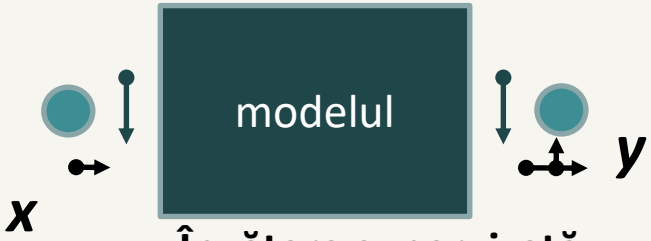
Învatare semi-supervizată

(niste date au etichete lipsa)

Învatare nesupervizata (datele **nu au etichete**)

“reinforcement learning” –
datale sunt acumulate pe parcurs

Invatare supervizata

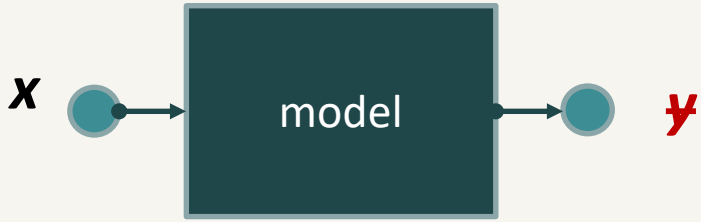


Învățare supervizată
(toate datele **au etichete**)

Recunoasterea fetelor :
Toate pozele din baza de date au etichete (stim cine e in poza)

Atunci sistemul nostru poate invata explicit ce il caracterizeaza pe Mihai Viteazul (e.g. pistrui)

Învățare nesupervizată



unsupervised learning
(datele nu **au etichete**)

Recunoașterea fetelor :
Imaginile de antrenament contin fete, dar nu stim ale cui.

Sistemul nu va invata explicit ce il diferentiaza pe Mihai Viteazul de restul lumii

Totusi sistemul poate invata ce e caracteristic unei fete

Invațare semi-supervizata

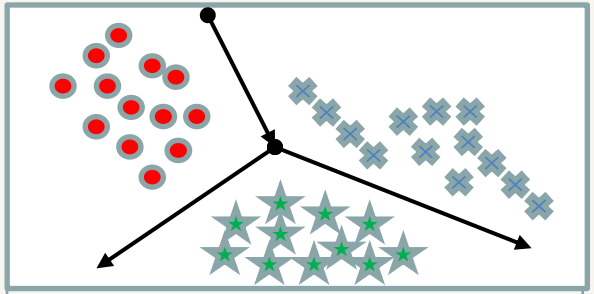
semi-supervised learning

(doar anumite date au etichete)

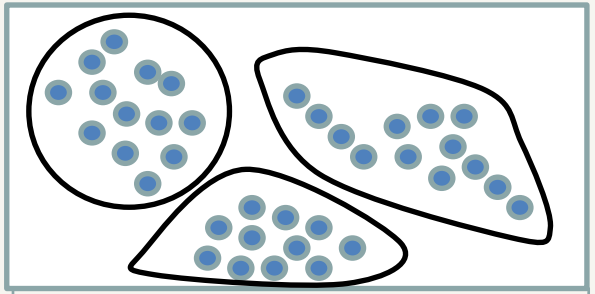
Recunoasterea fetelor:
Avem o baza de date in care anumite poze sunt nominale (stim cine e persoana), iar altele stim doar ca sontin o persoana

Putem folosi toate imaginile, invatand din cele fara etichete ce inseamna o fata iar din cele etichetate ce caracterizeaza o anumita persoana

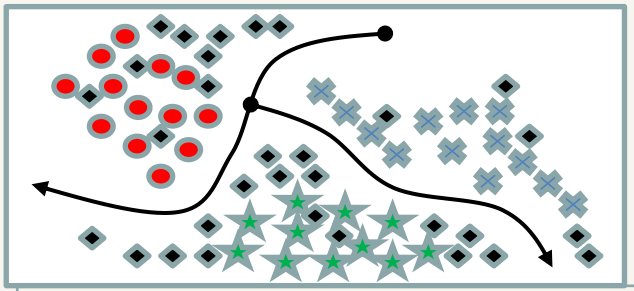
Variante de invatare



Invatare Supervizata



Invatare nesupervizata
(clustering)



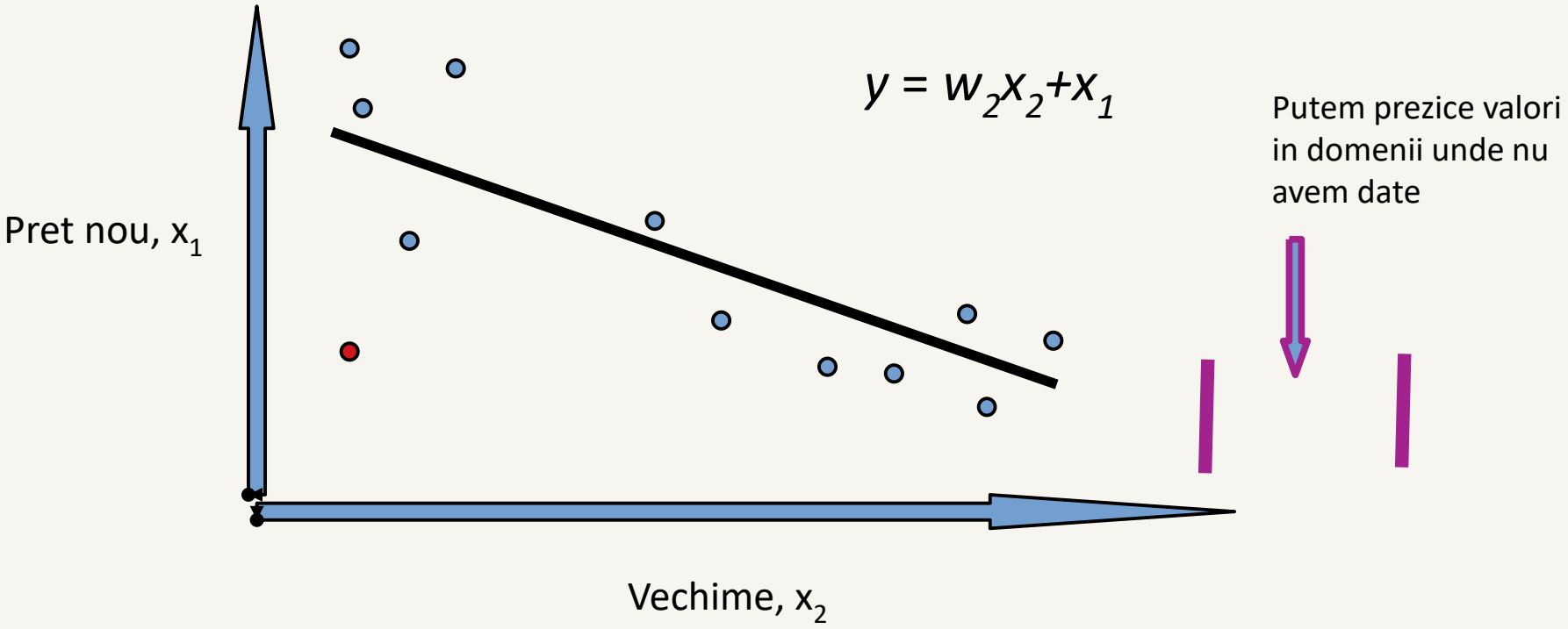
Invatare Semi-supervizata

Antrenare si testare

- Antrenarea este off-line:
  - Sistemul este in pregatire si se cauta valorile optime ale parametrilor
  - La terminarea antrenarii avem un se de valori ale parametrilor.
  - Daca parametri sunt cunoscuti, o functie parametrica devine perfect determinista
- Testarea este on-line:
  - Fiind dat setul de parametri de la antrenare, sistemul functioneaza:
  - Adica fiind date exemple noi, el prezice (estimeaza) iesirea

De ce “Invatare”?

- “Machine learning” inseamna sa programam calculatoare sa optimizeze un criteriu de performanta folosind exemple sau experienta (din trecut).
- Nu e nevoie sa “invatam” sa calculam salarii. Formula e fixa
- Invatarea are sens cand:
  - Experienta umana nu exista (e.g. sa navigam pe Marte),
  - Oamenii nu sunt in stare sa isi explice experienta (e.g. recunoasterea vorbirii)
  - Problema este prea complicata pentru ca un singur om să o rezolve
  - etc


$y = w_2x_2+x_1$
Pret nou, x_1
Vechime, x_2
Putem prezice valori in domenii unde nu avem date

Formalizare

- Vectorii de date sunt $\mathbf{x}_i$
- Etichetele sunt y_i
- Densitatea de probabilitate $P_{data}(x,y)$

Sistemul antrenabil este decris de setul de parametri

$$\boldsymbol{\theta} = \{\theta_1, \theta_2, \theta_3, \ldots.\}$$

  - Invatarea inseamna sa gasim setul $\boldsymbol{\theta}$, astfel incat legatura intre $\mathbf{x}$ si y sa fie optimala

Ipotezele IID

- Datele sunt descrise de densitatea de probabilitate:

$$x_i, y_i \sim p_{\text{data}}(x, y),\ i = \overline{1, n}$$

- Ipoteza 1:
  - Exemplele sunt extrase **independent** unul de altul
  - De ce? Legatura (dependenta) necesita modele prea complicate (e.g. Importanta diferita).
- Ipoteza 2:
  - Toate datele sunt **identic distribuite**, adica extrase din aceeasi distrubutie
  - De ce? Nu are nici un sens invatarea daca, de exemplu testul este extras din alta distributie

Functia obiectiv

- Este functia care leagă parametri **θ** de date
- Procesul de invatare este un proces de optimizare:
  - Minimizarea unei erori
  - Maximizarea unei probabilități sau corelații

In general, se cauta să se minimizeze o **functie cost (loss)** peste setul de antrenare:

$$\Theta^* = \arg\min_{\theta} L(\theta)$$

Exemplu: Predicție - Regresie

- Pretul unui apartament vechi
- *x* : atribute ale apartamentului (pret original, vechime)

y : pret curent

$y = g(x \mid \vartheta)$ Pt un ϑ dat si un x ales cat e y

g ( ) model - linear,

Θ parametrii: $\vartheta = \{w_1\}$

Cost: Eroare patratica medie (MSE) in pretul real si cel estimat

$$\hat{y}_i = wx_i + w_0$$

Functie MSE:

- Zero – pretul prezis este identic cu cel real
- Mica – in general pretul prezis este aproape de cel real
- Mare – pretul prezis este departe de cel real

Sisteme de clasificare

In functie de modul in care sunt legati parametri si functionarea lor, avem diferite sisteme de invatare:

- k-Nearest Neighbor – k-NN (cel mai apropiat vecin)
- Perceptron – multi layer (strat) perceptron (MLP) – artificial neural networks (ANN)
- Regresie Lineara (LR). Logistica pt clasificare
- Retele convolutionale – convolutional neural networks (CNN) – deep networks
- Masini cu vectori suport - Support vector machines (SVM)
- Arbori de clasificare si decizie - Decision and classification tree
- Masini cu invatare extrema - Extreme learning machine (ELM)
- Metode de tip ansamblu: bagging, boosting, aggregating

Care e cel mai bun?

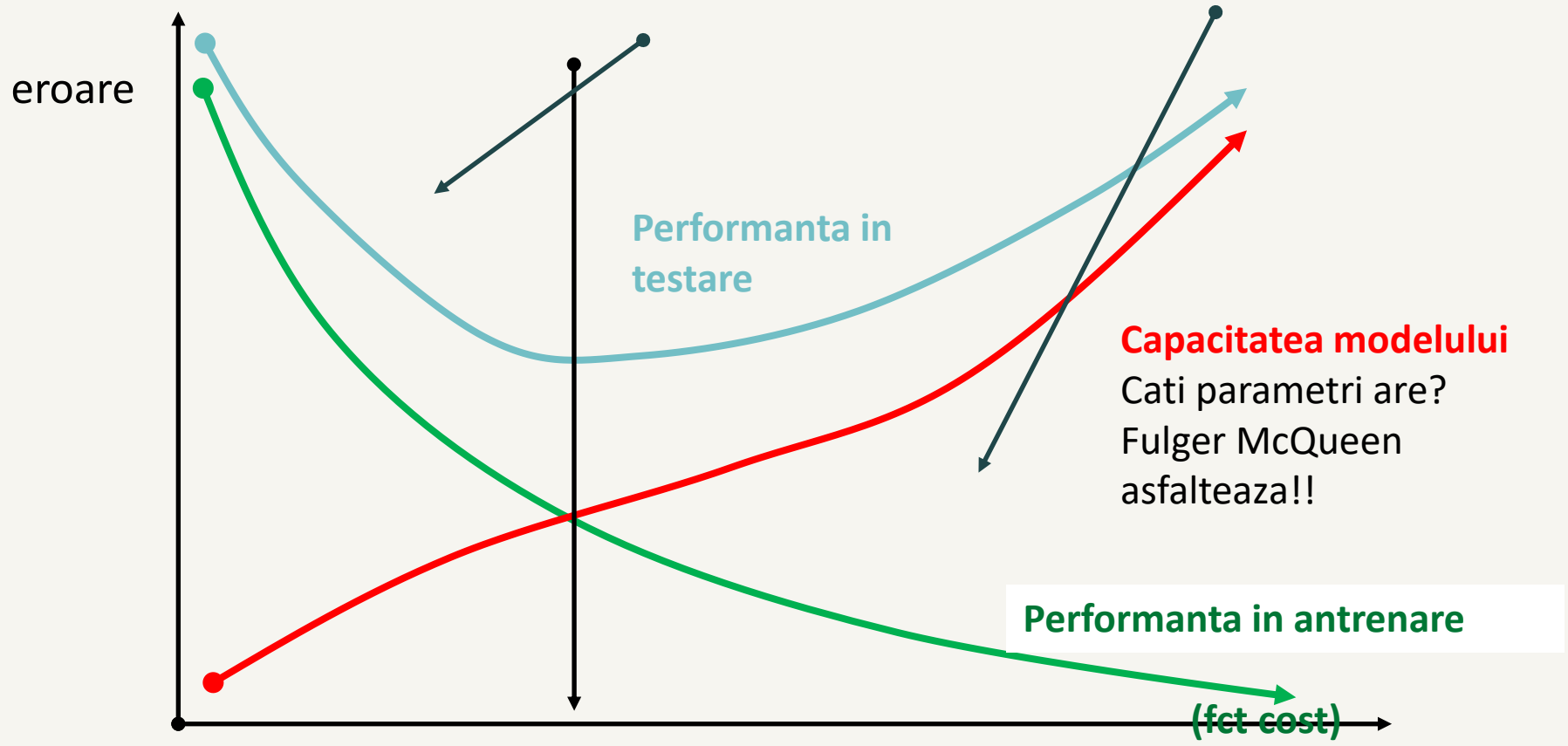
Teorem "no free lunch" *(Wolpert, 1996)* :

1. Rezultat demonstrat matematic
2. Consecinta: Pentru orice sistem care merge bine pe o anumita problemă (topologie a spațiului/ densitate de probabilitate), va exista alta problema pe care merge prost

Si atunci noi ce mai facem?

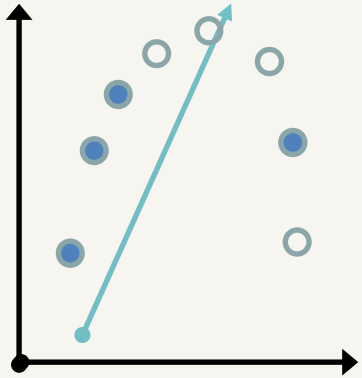
1. Cautam sisteme care merg bine pe categorii largi de probleme
2. Random Forest, SVM, Deep Learning

Subinvatare si Suprainvatare

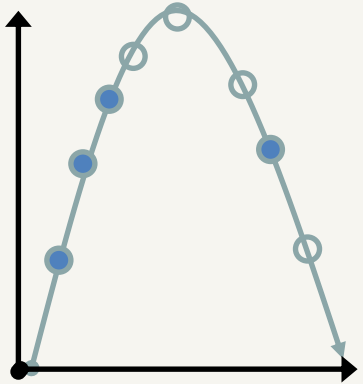


Subînvățare și Supraînvățare

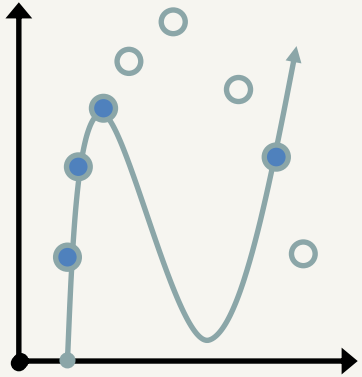
- **Subînvățare**: nu putem sa gasim un model care sa se potriveasca bine pe datele de antrenare (eroare mare pe setul de antrenare)
- **Supraînvățare**: nu putem gasi un model care sa generalizeze bine pe setul de testare (eroare mica pe setul de antrenare, eroare mare pe setul de test). In fapt memoreaza datele de antrenare.



Underfitting
(subinvățare)



appropriate capacity
(model de capacitate potrivita)



Overfitting
(Supraînvățare)