

# INTELIGENȚĂ , ARTIFICIALĂ



**Sisteme inteligente**

Învățare semi-supervizată-

**Laura Dioșan**

# Sumar

---

## A. Scurtă introducere în Inteligența Artificială (IA)

## B. Rezolvarea problemelor prin căutare

- Definirea problemelor de căutare
- Strategii de căutare
  - Strategii de căutare neinformate
  - Strategii de căutare informate
  - Strategii de căutare locale (Hill Climbing, Simulated Annealing, Tabu Search, Algoritmi evolutivi, PSO, ACO)
  - Strategii de căutare adversială

## C. Sisteme inteligente

- Sisteme care învață singure
  - Arbori de decizie
  - Rețele neuronale artificiale
  - Algoritmi evolutivi
  - Algoritmi de învățare semi-supervizată
- Sisteme bazate pe reguli
- Sisteme hibride

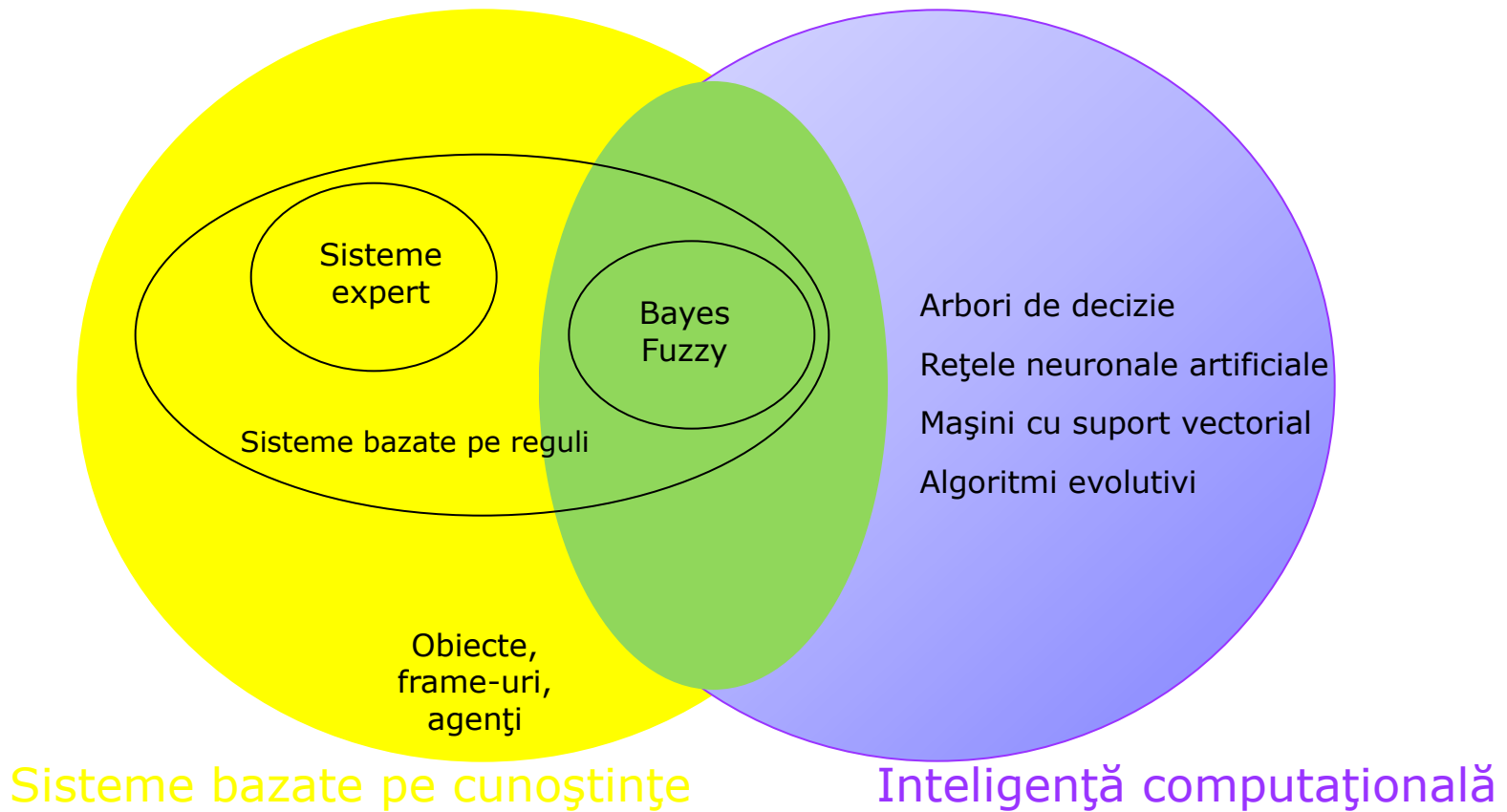
# Materiale de citit și legături utile

---

- ❑ capitolul 15 din *C. Groșan, A. Abraham, Intelligent Systems: A Modern Approach, Springer, 2011*
- ❑ Capitolul 9 din *T. M. Mitchell, Machine Learning, McGraw-Hill Science, 1997*
- ❑ Documentele din directorul *svm*

# Sisteme inteligente

---



# Învățare semi-supervizată

---

- De ce?
- Problema învățării
- Algoritmi (câțiva)

# Învățare semi-supervizată

---

## □ De ce?

- Datele ne-adnotate sunt ieftine
- Datele adnotate sunt greu de obținut/creat
  - Adnotarea de către oameni este o activitate plictisitoare
  - Etichetarea necesită
    - profesioniști/experti în domeniul respectiv
    - Dispozitive speciale
  - Studentul este în vacanță :D
- Exemple greu de etichetat
  - Analiza vorbirii
    - Transcrierea conversațiilor (telefonice)
    - O oră de vorbit = 400 ore de adnotat
  - Parsarea limbajului natural
- Exemple mai puțin dificile
  - Categorizare de imagini – google image

# Învățare semi-supervizată

---

## □ Problema învățării

- Antrenarea unor modele utilizând atât date adnotate, cât și date ne-adnotate

### ■ Date de antrenare:

- Etichetate (adnotate)  $(X_l, Y_l) = \{(x_{1:l}, y_{1:l})\}$
- Ne-etichetate  $X_u = \{x_{l+1:n}\}$  cu  $l \ll n$

### ■ Model

- $f : X \rightarrow Y$

### ■ Date de testare:

- $X_t = \{x_{n+1:\dots}\}$

# Învățare semi-supervizată

---

## □ Algoritmi

- Auto-antrenare
- Modele generative
- Mașini cu Suport Vectorial semi-supervizate
- Algoritmi bazați pe grafe



# Învățare semi-supervizată

---

## □ Algoritmi: Auto-antrenare (self-training)

### ■ Ideea de bază

1. Antrenarea modelului  $f$  pe datele  $(X_l, Y_l)$
2. Folosirea modelului pentru a eticheta un  $x \in X_u$
3. Adăugarea  $(x, f(x))$  la setul de date etichetate
4. Repetarea pașilor 1-3

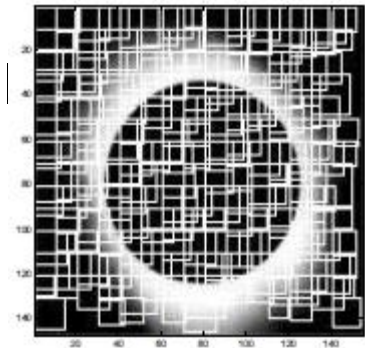
# Învățare semi-supervizată

## □ Algoritmi: Auto-antrenare

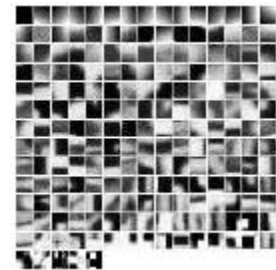
### ■ Exemple

#### □ Clasificarea imaginilor

- O imagine este împărțită în mai multe regiuni (normalizate)



- Se definește un dicționar de "cuvinte vizuale" (centrozii ai clusterizării)



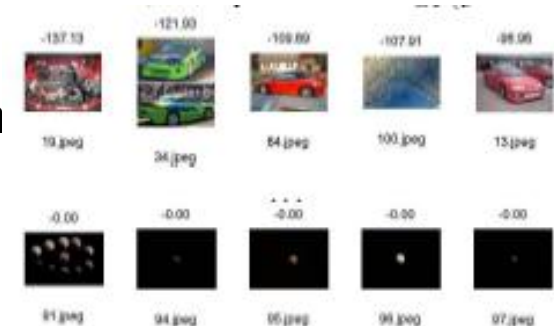
- Se reprezintă fiecare regiune prin indexul celui mai apropiat "cuvânt vizual"

# Învățare semi-supervizată

## □ Algoritmi: Auto-antrenare

### ■ Exemple - Clasificarea imaginilor

1. Se antrenează un clasificator pe imagini adnotate



2. Se clasifică imaginile ne-etichetate (sortându-se pe baza unei măsuri de încredere)



3. Cele mai reprezentative (de încredere) imagini (împreună cu etichetele lor) se adaugă în setul de imagini etichetate



4. Se repetă pașii 1-3

# Învățare semi-supervizată

---

## □ Algoritmi: Auto-antrenare

### ■ Avantaje

- Metodă simplă
- Metodă de tip wrapper, aplicată unor clasificatori existenți
- Folosită des în task-uri reale (NLP)

### ■ Dezavantaje

- Erorile timpurii pot fi consolidate și propagate ușor
  - Eliminarea etichetei unui exemplu cu încrederea sub un anumit prag
- Puține informații despre convergență
  - În unele cazuri, auto-antrenare = maximizarea așteptărilor

# Învățare semi-supervizată

---

## □ Algoritmi: Modele generative

### ■ Ideea de bază

- Date etichetate  $(X_l, Y_l)$
- Se presupune că datele dintr-o clasă respectă o distribuție Gaussiană
  
- Clusterizarea datelor  $X_l$  și  $X_u$
- Etichetarea datelor dintr-un cluster cu eticheta datelor etichetate majoritare

# Învățare semi-supervizată

---

- Algoritmi: Mașini cu Suport Vectorial semi-supervizate (S3VMs = Transductive SVMs)
  - Ideea de bază
    - Maximizarea marginilor datelor ne-etichetate
    - Se enumeră toate cele  $k^u$  posibilități de a eticheta datele  $X_u$
    - Se construiește un SVM clasic pentru fiecare posibilitate
    - Se alege SVM cu cea mai largă margine

# Învățare semi-supervizată

---

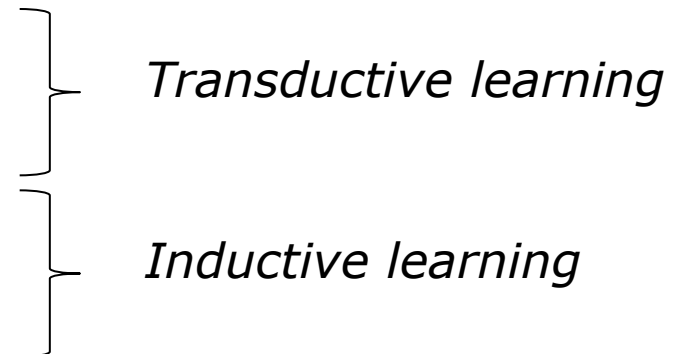
## □ Algoritmi: Algoritmi bazați pe grafe

### ■ Ideea de bază

- Noduri:  $X_l \cup X_u$
- Muchii: ponderi pentru similaritatea diferitelor atribute ale nodurilor
  - K-cel mai apropiat vecin (ponderi booleene)
  - Grafe complete (ponderi invers proporționale cu distanța între noduri)
- Stabilirea similarității pe toate căile

### ■ Algoritmi

- *Mincut*
- *Harmonic functions*
  
- Consistență locală și globală
- Regularizare manifold



# Învățare semi-supervizată

---

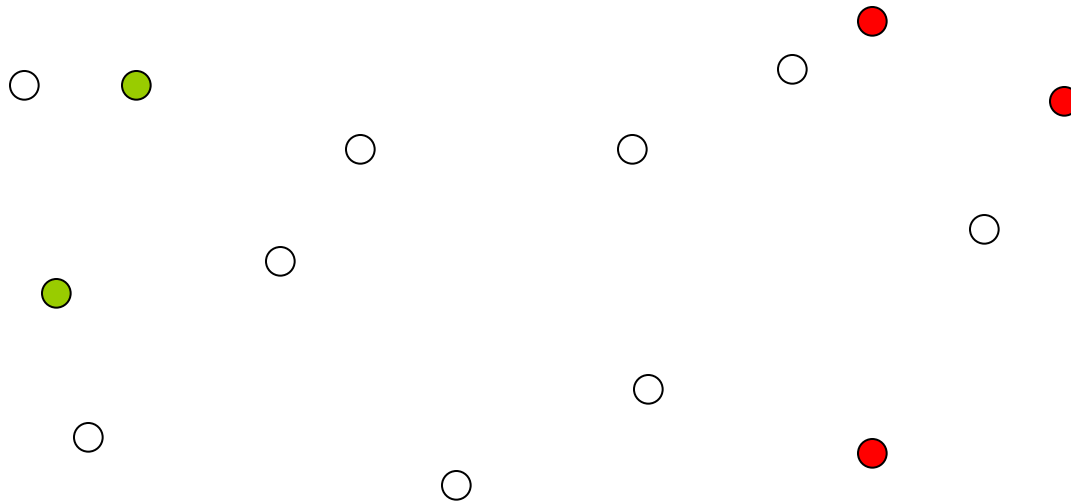
- Algoritmi: Algoritmi bazați pe grafe - *Mincut*
  - Pp că exemplele similare trebuie să fie adnotate (etichetate) similar



# Învățare semi-supervizată

---

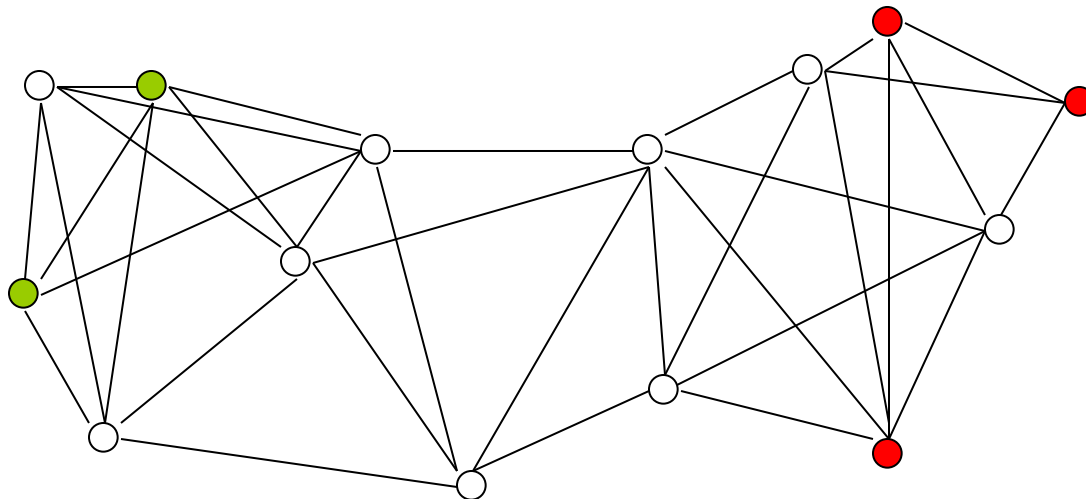
- Algoritmi: Algoritmi bazați pe grafe - *Mincut*
  - Plecăm de la un set de date  $((X_l, Y_l), X_+)$



# Învățare semi-supervizată

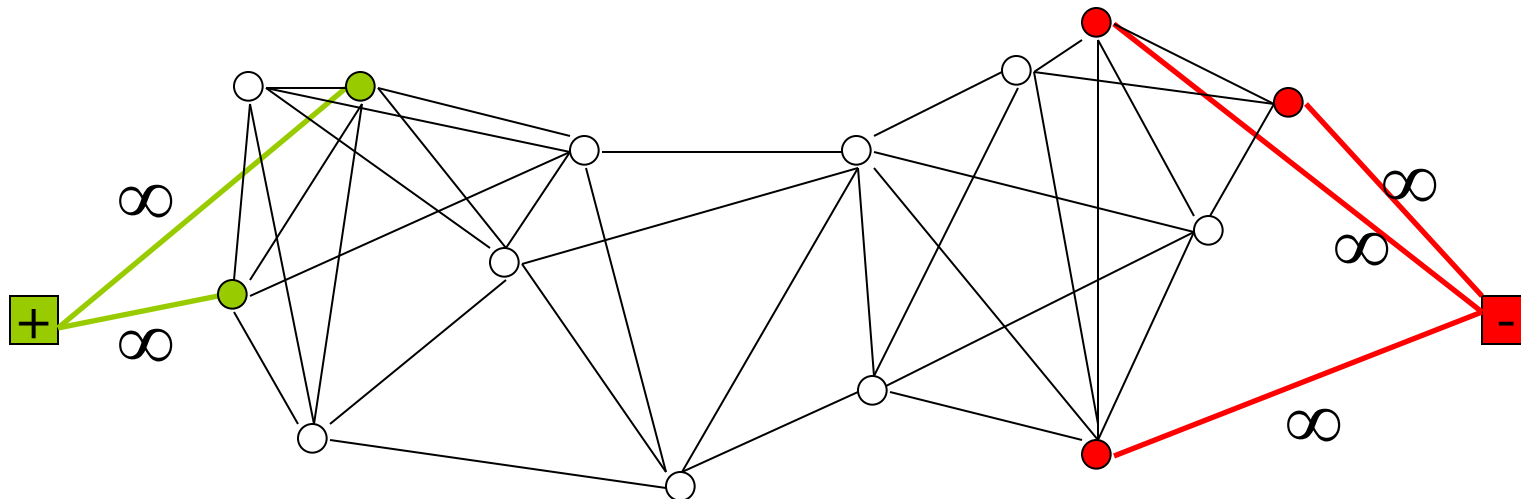
---

- Algoritmi: Algoritmi bazați pe grafe - *Mincut*
  - Plecăm de la un set de date  $((X_I, Y_I), X_U)$
  - Construim un graf (ponderat/neponderat)



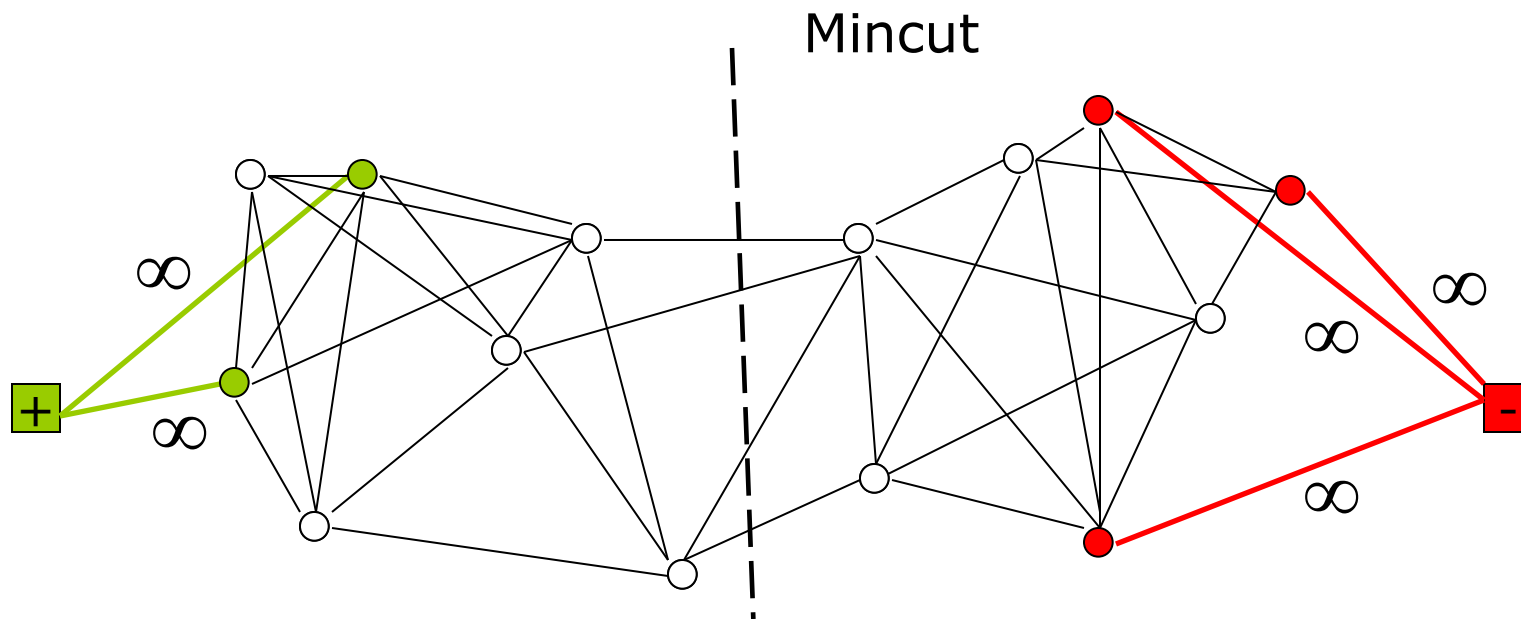
# Învățare semi-supervizată

- Algoritmi: Algoritmi bazați pe grafe - *Mincut*
  - Plecăm de la un set de date  $((X_I, Y_I), X_U)$
  - Construim un graf (ponderat/neponderat)
  - Adăugăm super-noduri auxiliare



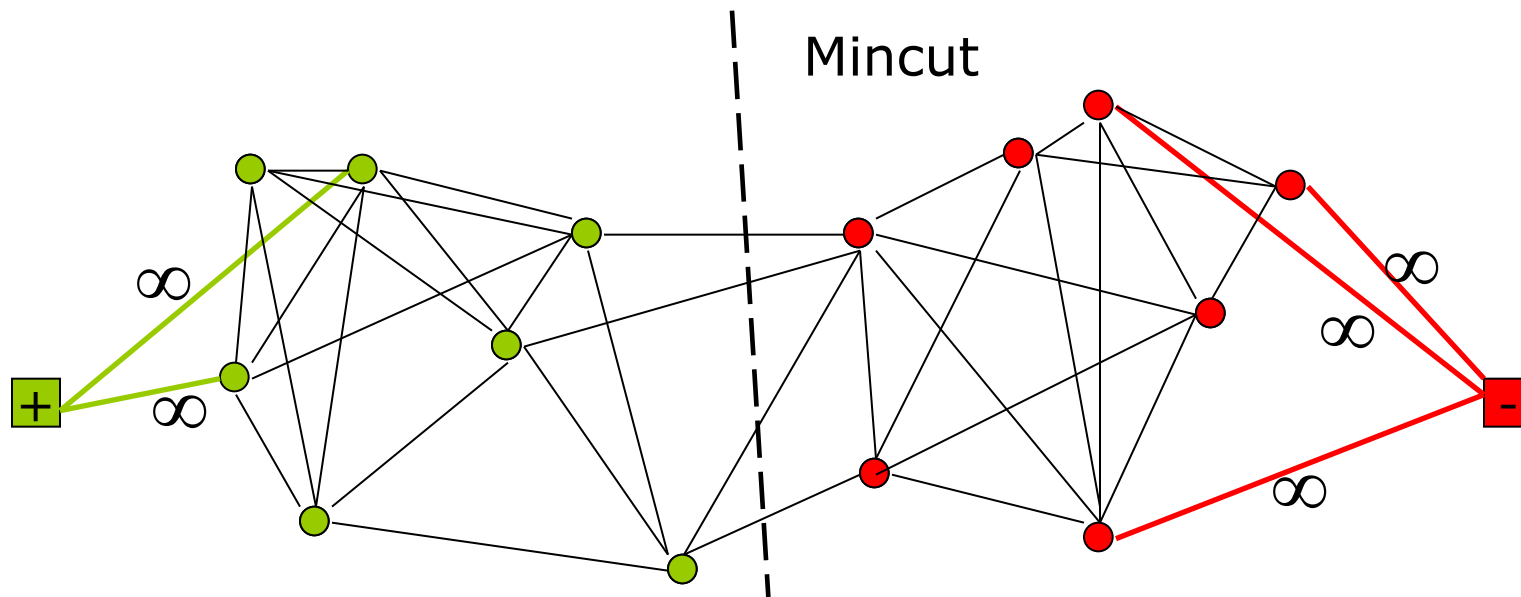
# Învățare semi-supervizată

- Algoritmi: Algoritmi bazați pe grafe - *Mincut*
  - Plecăm de la un set de date  $((X_l, Y_l), X_u)$
  - Construim un graf (ponderat/neponderat)
  - Adăugăm super-noduri auxiliare
  - Obținem o tăietură minimă



# Învățare semi-supervizată

- Algoritmi: Algoritmi bazați pe grafe - *Mincut*
  - Plecăm de la un set de date  $((X_l, Y_l), X_u)$
  - Construim un graf (ponderat/neponderat)
  - Adăugăm super-noduri auxiliare
  - Obținem o tăietură minimă
  - Clasificăm



# Învățare semi-supervizată

---

- Algoritmi: Algoritmi bazați pe grafe – *Mincut*
  - Construcția grafului – metode
    - **k-NN**
      - Graful poate să nu aibă tăieturi echilibrate
      - Cum se învață  $k$ ?
    - Conectarea tuturor punctelor sub o distanță prag  $\delta$ 
      - Pot apărea componente de-conectate
      - Cum se învață pragul  $\delta$ ?
    - Arbore de acoperire minimă
      - Fără parametri
      - Rezultă grafe conectate și rare
      - Funcționează bine pe multe dintre date

# Recapitulare

---



- Sisteme care învață singure (SIS)
  - Învățare semi-supervizată
    - Auto-antrenare
    - Modele generative
    - Mașini cu Suport Vectorial semi-supervizate
    - Algoritmi bazați pe grafe

# Cursul următor

---

## A. Scurtă introducere în Inteligența Artificială (IA)

## B. Rezolvarea problemelor prin căutare

- Definirea problemelor de căutare
- Strategii de căutare
  - Strategii de căutare neinformate
  - Strategii de căutare informate
  - Strategii de căutare locale (Hill Climbing, Simulated Annealing, Tabu Search, Algoritmi evolutivi, PSO, ACO)
  - Strategii de căutare adversarială

## C. Sisteme inteligente

- Sisteme care învață singure
  - Arbori de decizie
  - Rețele neuronale artificiale
  - Mașini cu suport vectorial
  - Algoritmi evolutivi
- Sisteme bazate pe reguli
- Sisteme hibride



---

□ Informațiile prezentate au fost colectate din diferite surse de pe internet, precum și din cursurile de inteligență artificială ținute în anii anteriori de către:

- Conf. Dr. Mihai Oltean – [www.cs.ubbcluj.ro/~moltean](http://www.cs.ubbcluj.ro/~moltean)
- Lect. Dr. Crina Groșan - [www.cs.ubbcluj.ro/~cgrosan](http://www.cs.ubbcluj.ro/~cgrosan)
- Prof. Dr. Horia F. Pop - [www.cs.ubbcluj.ro/~hfpop](http://www.cs.ubbcluj.ro/~hfpop)