# Rețele Bayesiene

## Teorie

### Evenimente

Un **eveniment** este un **subset al spațiului de probabilitate (sau spațiului fundamental)**. Spațiul de probabilitate (notat de obicei cu Ω) este **mulțimea tuturor rezultatelor posibile** ale unui experiment aleator.

#### Probabilități pentru evenimente combinate

##### Intersecția a două evenimente

**Intersecția a două evenimente** A și B este evenimentul care are loc dacă și numai dacă ambele evenimente A și B au loc simultan.

P(A∩B)=P(A)×P(B∣A) sau echivalent, P(A∩B)=P(B)×P(A∣B)

unde P(B∣A) este probabilitatea ca evenimentul B să aibă loc, dat fiind că evenimentul A a avut deja loc (probabilitatea condiționată), și P(A∣B) este probabilitatea ca evenimentul A să aibă loc, dat fiind că evenimentul B a avut deja loc.

#### Reuniunea a două evenimente

**Reuniunea a două evenimente** A și B este evenimentul care are loc dacă și numai dacă cel puțin unul dintre evenimentele A sau B (sau ambele) are loc.

P(A∪B)=P(A)+P(B)−P(A∩B)

Această formulă este valabilă pentru orice două evenimente A și B, fie ele independente sau dependente, compatibile sau incompatibile. Scădem P(A∩B) pentru a evita numărarea dublă a cazurilor în care ambele evenimente au loc.

#### Evenimente mutual exclusive

Evenimentele **mutual exclusive (sau disjuncte)** sunt evenimente care nu pot avea loc simultan (dacă un eveniment se întâmplă, celelalte evenimente mutual exclusive nu se pot întâmpla)

Observații:

* **Intersecția** a două evenimente mutual exclusive este **mulțimea vidă**

P(A și B) = P(A ∩ B) =0

* Probabilitatea ca oricare dintre evenimentele mutual exclusive să aibă loc este suma probabilităților individuale ale evenimentelor.

**Pentru două evenimente mutual exclusive A și B:**

P(A sau B) = P(A ∪ B) = P(A) + P(B)

**Pentru n evenimente mutual exclusive A₁, A₂, ..., Aₙ:**

P(A₁ sau A₂ sau ... sau Aₙ) = P(A₁ ∪ A₂ ∪ ... ∪ Aₙ) = P(A₁) + P(A₂) + ... + P(Aₙ)

#### Evenimente independente

Două evenimente sunt considerate **independente** dacă realizarea unuia dintre ele nu influențează probabilitatea realizării celuilalt. Cu alte cuvinte, faptul că un eveniment a avut loc sau nu, nu oferă nicio informație suplimentară despre șansele ca celălalt eveniment să se întâmple.

**Două evenimente A și B sunt independente dacă și numai dacă probabilitatea intersecției lor** (adică probabilitatea ca ambele evenimente să aibă loc simultan) **este egală cu produsul probabilităților lor individuale:**

P(A∩B)=P(A)×P(B)

### Variabile aleatorii

O **variabilă aleatorie** este o funcție care asociază un număr real fiecărui rezultat posibil al unui experiment aleator (este o variabilă ale cărei valori sunt determinate de un proces aleator).

Există două tipuri principale de variabile aleatoare:

* **Variabile aleatorii discrete:** Aceste variabile pot lua doar un număr finit sau numărabil de valori distincte. Exemple includ numărul de bile de o anumită culoare extrase dintr-o urnă cu bile de multiple culori. Numărul de dăți în care s-a obținut o anumită fațetă a unui zar în cadrul a N\_A aruncări.
* **Variabile aleatorii continue:** Aceste variabile pot lua orice valoare dintr-un interval dat. Exemple includ înălțimea unei persoane, temperatura sau timpul necesar pentru a finaliza o acțiune.

Fiecare valoare pe care o variabilă aleatorie o poate lua are o anumită probabilitate de a se întâmpla.

### Distribuții de probabilitate

O **distribuție de probabilitate** este o funcție matematică care descrie probabilitatea fiecărei valori posibile pe care o variabilă aleatoare o poate lua.

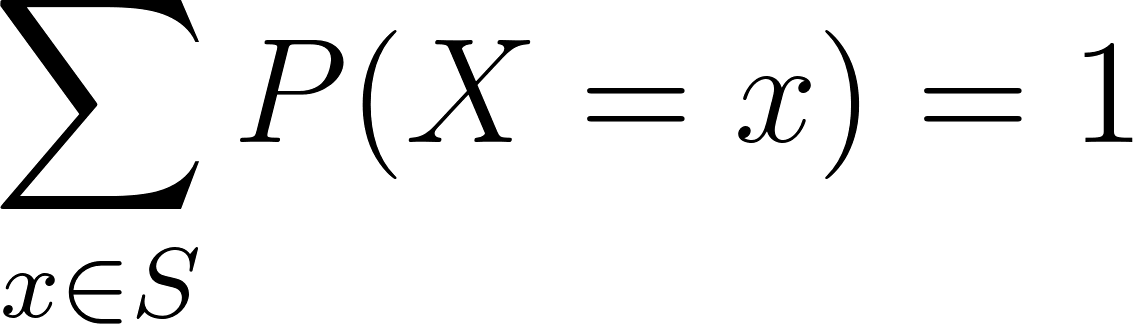
Pentru variabilele discrete, distribuția de probabilitate este adesea exprimată printr-o funcție de masă de probabilitate (PMF), care dă probabilitatea fiecărei valori posibile. Pentru variabilele continue, distribuția de probabilitate este descrisă de o funcție de densitate de probabilitate (PDF), astfel încât probabilitatea ca variabila să cadă într-un anumit interval este dată de integrala PDF-ului pe acel interval.

Distribuții de probabilitate discrete

Funcția de masă de probabilitate (PMF), este notată de obicei cu P(X=x) sau f(x).

Pentru fiecare valoare posibilă x a variabilei aleatoare discrete X, PMF-ul dă probabilitatea ca X să fie egal cu acea valoare: P(X=x)=px​.

PMF-ul trebuie să satisfacă două condiții:

* 0≤P(X=x)≤1 pentru toate valorile x.
*  , unde suma este luată peste toate valorile posibile din spațiul eșantion S (, adică setul tuturor valorilor posibile pe care le poate lua variabila aleatoare X).

### Structura rețelelor Bayesiene

O rețea Bayesiană (numită și rețea de încredere) este un model grafic probabilistic care reprezintă un set de variabile aleatorii și dependențele condiționale dintre ele printr-un graf aciclic orientat (*DAG - directed acyclic graph*).

**Componentele principale ale unei rețele Bayesiene sunt:**

1. **Nodurile:** Fiecare nod din graf reprezintă o variabilă aleatorie. Aceste variabile pot fi discrete (cu un număr finit de stări, cum ar fi "plouă" cu valorile posibile "da" sau "nu") sau continue (cu un număr infinit de valori posibile, cum ar fi temperatura).
2. **Arce (sau săgeți):** Arcele orientate conectează nodurile și reprezintă dependențe cauzale sau probabilistice directe între variabile. Dacă există un arc de la nodul A la nodul B, se interpretează că A are o influență directă asupra lui B sau că există o dependență probabilistică directă între A și B. Direcția săgeții nu implică neapărat o relație cauzală în sens strict, dar este adesea modelată ca atare.
3. **Probabilități Condiționale:** Fiecare nod care are părinți (noduri din care pleacă arce către el) este asociat cu o tabelă de probabilități condiționale (CPT - Conditional Probability Table). Această tabelă specifică distribuția de probabilitate a nodului pentru fiecare combinație posibilă de stări ale părinților săi. Pentru un nod fără părinți, se specifică distribuția de probabilitate marginală (probabilitatea fiecărei stări a variabilei respective).

**Suportul cauzal** al unei variabile aleatorii (nod în topologie) se referă la evidența sau informația care influențează probabilitatea unei variabile prin intermediul relațiilor cauzale directe sau indirecte reprezentate în rețea. Acesta implică un raționament de sus în jos (de la cauză la efect) sau în lanț prin legăturile cauzale.

**Suportul probatoriu** al unei variabile aleatorii (nod în topologie) se referă la raționamentul de jos în sus (de la variabilele observate mergând din părinte în părinte spre nosul curent)

### Funcționarea rețelelor Bayesiane

Ideea fundamentală din spatele rețelelor Bayesiene este de a utiliza teorema lui Bayes pentru a calcula probabilități condiționale. Teorema lui Bayes este:

P(A∣B)=P(B)P(B∣A)⋅P(A)​

unde:

* P(A∣B) este probabilitatea evenimentului A dat fiind că evenimentul B a avut loc (probabilitatea a posteriori).
* P(B∣A) este probabilitatea evenimentului B dat fiind că evenimentul A a avut loc (verosimilitatea).
* P(A) este probabilitatea a priori a evenimentului A.
* P(B) este probabilitatea a priori a evenimentului B (probabilitatea totală).

Într-o rețea Bayesiană, putem introduce dovezi (observații despre stările anumitor variabile) și apoi putem actualiza probabilitățile celorlalte variabile din rețea, calculând probabilitățile a posteriori. Acest proces se numește **inferență Bayesiană**.

**Exemplu Simplu:**

Să considerăm o rețea Bayesiană simplă cu trei variabile:

* **Ploaie (P):** Poate fi "Da" sau "Nu".
* **Udat (U):** Iarba este "Udată" sau "Neudată".
* **Stropitoare (S):** Stropitoarea este "Pornită" sau "Oprită".

Putem modela următoarele dependențe:

* Ploaia are o influență asupra faptului dacă iarba este udată.
* Stropitoarea are o influență asupra faptului dacă iarba este udată.
* Ploaia și stropitoarea sunt independente una de cealaltă (în acest model simplificat).

Graficul ar arăta astfel:

P ---> U <--- S

Fiecare nod ar avea asociată o tabelă de probabilități:

* **Ploaie (P):** P(P=Da), P(P=Nu) (probabilități marginale).
* **Stropitoare (S):** P(S=Pornită), P(S=Oprită) (probabilități marginale).
* **Udat (U):** P(U=Udată∣P,S) pentru toate combinațiile posibile ale stărilor lui P și S (Da/Nu pentru P, Pornită/Oprită pentru S). Aceasta ar include P(U=Udata˘∣P=Da,S=Pornita˘), P(U=Udată∣P=Da,S=Oprită), P(U=Udată∣P=Nu,S=Pornită), P(U=Udată∣P=Nu,S=Oprită), și probabilitățile complementare pentru P(U=Neudată∣P,S).

Dacă observăm că iarba este "Udată" (U=Udată), putem folosi inferența Bayesiană pentru a calcula probabilitatea ca a plouat (P=Da) sau ca stropitoarea a fost pornită (S=Pornită), ținând cont de probabilitățile a priori și de dependențele modelate.

### Poliarbori

Un **poliarbore** (*polytree; singly connected Bayesian network*) este un tip specific de graf orientat aciclic (DAG) utilizat pentru a reprezenta o rețea Bayesiană cu o structură arborescentă relaxată.

Caracteristica definitorie a unui poliarbore este că **între oricare două noduri din graf există cel mult o singură cale (neorientată)**.

Prin urmare, deși un nod poate avea mai mulți părinți (spre deosebire de un arbore Bayesian strict, unde fiecare nod are cel mult un părinte), nu există cicluri în graf și nu există două căi distincte (independente) între oricare două noduri.

### Drumuri blocate condiționat

În cadrul unei rețele Bayesiene, un **drum blocat condiționat** (*conditionally blocked path*) reprezintă o secvență de noduri conectate prin arce (într-o anumită direcție), unde transmiterea influenței probabilistice (sau a dependenței) între două noduri de la capetele drumului este oprită sau blocată de starea cunoscută (condiționată) a unuia sau mai multor noduri intermediare de pe acel drum.

Considerăm **o mulțime de noduri E observate** (cunoscute, deci considerate *"evidence"*). Există trei configurații de bază care pot bloca un drum condiționat:

**1. Structura X → Z ← Y**

* **Blocare:** Un drum între X și Y este **blocat condiționat de E dacă Z nu este observat (adică, starea lui nu este cunoscută) sau oricare dintre descendenții lui Z nu sunt observați (nici Z și nici descendenții nu aparțin lui E).**
* **Intuiție:** Dacă nu știm nimic despre Z, atunci X și Y pot fi marginal independente. Însă, dacă observăm starea lui Z, atunci informația despre X poate oferi informații despre Y (și invers), deoarece ambele influențează Z. Dacă un descendent al lui Z este observat, acesta oferă informații indirecte despre Z, activând astfel dependența între X și Y.
* **Exemplu:**
  + X = Lipsă de somn
  + Y = Stres
  + Z = Durere de cap. Dacă nu știm dacă o persoană are dureri de cap, lipsa de somn și stresul pot fi independente în influența lor asupra altor variabile. Însă, dacă știm că persoana are dureri de cap, atunci a ști că a avut puțin somn ar putea reduce probabilitatea ca stresul să fie cauza principală, introducând o dependență (negativă) între lipsa de somn și stres, condiționată pe durerea de cap.

**2. Structura X→Z→Y sau X←Z←Y**

* **Blocare:** Un drum între X și Y este **blocat condiționat de E dacă Z este observat (Z ∈ E).**
* **Intuiție:** Dacă știm starea lui Z, atunci X devine independent de Y (și invers), deoarece toată influența dintre ele trece prin Z, iar starea lui Z este fixată.
* **Exemplu (pentru X→Z→Y):**
  + X = Ploaie
  + Z = Trotuar ud
  + Y = Alunecare. Dacă știm că trotuarul este ud, atunci dacă a plouat sau nu devine irelevant pentru probabilitatea de a aluneca (în măsura în care ploaia influențează alunecarea doar prin faptul că face trotuarul ud). X și Y devin independente condiționate pe Z.

**3. Structura** X←Z→Y

* **Blocare:** Un drum între X și Y este **blocat condiționat pe Z dacă Z este observat (Z ∈ E).**
* **Intuiție:** Z este o cauză comună pentru X și Y. Dacă știm starea lui Z, atunci informația despre X nu mai oferă informații suplimentare despre Y (și invers), deoarece cauza comună este cunoscută.
* **Exemplu:**
  + Z = Defect de fabricație
  + X = Produs defect (tip 1)
  + Y = Produs defect (tip 2). Ambele tipuri de defecte sunt cauzate de același defect de fabricație. Dacă știm că a existat un defect de fabricație, atunci observarea unui produs defect de tip 1 nu ne spune nimic nou despre probabilitatea ca un alt produs să aibă defect de tip 2 (deoarece amândouă sunt deja explicate de defectul de fabricație cunoscut). X și Y devin independente condiționate pe Z.

### Independența condiționată

Două evenimente sau variabile aleatoare A și B sunt **independente condiționat** pe un al treilea eveniment sau variabilă aleatoare C dacă probabilitatea ca A să se întâmple (sau să ia o anumită valoare), dată fiind informația că B s-a întâmplat (sau a luat o anumită valoare), nu este influențată de faptul că C s-a întâmplat (sau a luat o anumită valoare).

Formal, pentru evenimente A, B, și C cu P(C)>0, A și B sunt independente condiționat de C dacă și numai dacă:

P(A∣B,C)=P(A∣C)

Sau, echivalent:

P(B∣A,C)=P(B∣C)

Sau, în termeni de probabilitate comună (să se întâmple și A și B):

P(A ∩ B | C) = P(A,B∣C)=P(A∣C)P(B∣C)

### D-separarea

D-separarea ne spune dacă fluxul de influență probabilistică (sau dependența) între două seturi de variabile este "blocat" de cunoașterea stării unui al treilea set de variabile. Dacă toate căile posibile de influență sunt blocate, atunci cele două seturi devin independente în contextul informației date.

Două noduri (sau seturi de noduri) X și Y sunt **d-separate** de un set de noduri E dacă toate căile neorientate între un nod din X și un nod din Y sunt "blocate" de setul E

**Două noduri X şi Y sunt independente condiţionat de E, dacă toate drumurile nedirecționate (ignorând sensul arcelor) de la X la Y sunt d-separate de E**

### Utilizarea rețelelor Bayesiene

Rețelele Bayesiene sunt instrumente puternice pentru modelarea și raționamentul în condiții de incertitudine. Ele sunt utilizate într-o gamă largă de aplicații, inclusiv:

1. **Diagnostic medical:**
   * Modelarea relațiilor dintre simptome, boli și factori de risc.
   * Calcularea probabilității unei anumite boli pe baza simptomelor observate.
   * Asistarea medicilor în luarea deciziilor de diagnostic și tratament.
2. **Sisteme de expert:**
   * Reprezentarea cunoștințelor experților într-un format probabilistic.
   * Realizarea de inferențe și recomandări pe baza dovezilor disponibile.
3. **Recunoaștere de pattern-uri și clasificare:**
   * Clasificarea imaginilor, textului sau a altor tipuri de date.
   * Modelarea probabilității ca un anumit pattern să aparțină unei anumite clase.
4. **Bioinformatică:**
   * Analiza rețelelor genetice și proteice.
   * Inferarea relațiilor dintre gene, proteine și boli.
5. **Finanțe:**
   * Modelarea riscului și a incertitudinii pe piețele financiare.
   * Predicția probabilității de default sau a altor evenimente financiare.
6. **Inginerie și fiabilitate:**
   * Analiza fiabilității sistemelor complexe.
   * Identificarea cauzelor probabile ale defecțiunilor.
7. **Inteligență artificială și învățare automată:**
   * Modelarea proceselor decizionale sub incertitudine.
   * Învățarea structurii rețelelor din date.
   * Construirea de agenți inteligenți care pot raționa și lua decizii în medii incerte.
8. **Procesarea limbajului natural:**
   * Modelarea dependențelor dintre cuvinte și fraze.
   * Realizarea de analiză semantică și înțelegere a textului.
9. **Viziune artificială:**
   * Înțelegerea scenelor vizuale prin modelarea relațiilor dintre obiecte și atribute.

### Avantajele Rețelelor Bayesiene

* **Claritate și interpretabilitate:** Structura grafică face ca relațiile dintre variabile să fie ușor de înțeles și vizualizat.
* **Manevrarea incertitudinii:** Oferă un cadru formal pentru a raționa sub incertitudine, utilizând probabilități.
* **Combinarea datelor cu cunoștințele anterioare:** Permite încorporarea cunoștințelor experților (sub formă de probabilități a priori și structură a rețelei) cu datele observate.
* **Capacitatea de a învăța din date:** Structura rețelei și/sau probabilitățile pot fi învățate automat din date statistice.
* **Inferență flexibilă:** Permite calcularea probabilității oricărei variabile (sau set de variabile) date fiind observații despre alte variabile.

### Dezavantajele Rețelelor Bayesiene

* **Complexitatea calculului:** Inferența în rețele Bayesiene complexe poate fi costisitoare din punct de vedere computațional (în special pentru rețele cu multe variabile și dependențe complexe).
* **Elicitarea probabilităților:** Obținerea probabilităților condiționale precise de la experți poate fi dificilă și subiectivă.
* **Învățarea structurii:** Învățarea structurii rețelei dintr-un set de date poate fi o problemă dificilă, în special când datele sunt limitate.

## Context exerciții

Pentru toate exercițiile de mai jos, fișierul de intrare are formatul descris mai jos.

**pentru fiecare nod** (cu părinți) avem:

* un rând care începe cu "nod:" urmat de identificatorul lui (un string care definește la ce se referă; pentru simplitate, vom considera stringul fără spații, cuvintele diferite din identificator fiind separate prin underline, cum ar fi citeste\_carte\_domeniu)
* un rând cu părinții nodului respectiv (adică identificatorii lui)
* urmat de 2n rânduri cu toate combinațiile de adevărat/fals pentru părinți si probabilitatea ca în acea situație valoarea nodului curent să fie adevărată. Fiecare astfel de rând începe cu valori binare (din setul: 1,0). Valoarea binară de pe poziția i corespunde valorii de adevăr pentru părintele de pe poziția i. Rândul se încheie cu probabilitatea condiționată ca nodul curent să fie adevărat pentru valorile de adevăr indicate ale părinților

Pentru fiecare nod fără părinți avem:

* un rând care începe cu "nod:" urmat de identificatorul lui
* un rând cu probabilitatea ca nodul să fie adevărat.

Pe penultimul rând, precedate de textul "observatii:" vor fi trecute variabilele observate și valoare lor de adevăr. Variabilele au formatul nume\_variabila:valoare\_adevar și sunt separate cu ";" urmat de spațiu.

Pe ultimul rând, precedat de textul "interogare:" va fi trecut identificatorul variabilei de interogare.

Exemplu de fișier de intrare:

nod: angajat

0.4

nod: vine\_la\_ore

angajat

1 0.2

0 0.7

nod: jocuri\_calculator

0.5

nod: planifica\_timp

0.4

nod: timp\_studiu\_acasa

jocuri\_calculator, planifica\_timp

1 1 0.5

1 0 0.1

0 1 0.95

0 0 0.3

nod: pasionat

0.35

nod: citeste\_carte\_domeniu

pasionat, timp\_studiu\_acasa

1 1 0.85

1 0 0.45

0 1 0.3

0 0 0.05

nod: face\_temele

vine\_la\_ore, timp\_studiu\_acasa

1 1 0.95

1 0 0.35

0 1 0.7

0 0 0.15

nod: promoveaza\_materia

face\_temele

1 0.9

0 0.4

observatii:

angajat:0; jocuri\_calculator:1; pasionat:1; planifica\_timp:0

interogare: promoveaza\_materia

Pentru topologia:

## 

## Exerciții

1) **[deja rezolvat]** Definiți o clasa Nod, care va reprezenta un nod din rețeaua Bayesianăcu câmpurile: identificator, parinti, fii, probabl. In constructor, vom defini pentru parinti si fii, ca valori implicite lista vidă. Proprietatea probabl, reprezintă probabilitatea ca variabila asociată nodului să fie adevărată și va avea ca valoare implicită None.

În plus, va exista și proprietatea tabel, în care va fi stocat tabelul de probabilități condiționate sub formă de listă de liste. Pentru fiecare listă, primele elemente vor reprezenta valori booleene corespunzătoare părinților nodului și ultima valoare va fi probabilitatea condiționată corespunzătoare nodului curent.

Va exista și proprietatea val (folosită pentru nodurile observate).

Se vor defini metodele:

* \_\_str\_\_ care afișează nodul cu probabilitatea asociată.
* \_\_repr\_\_ care afișează nodul cu toate informațiile ( toate proprietățile și valorile lor)

2) **[deja rezolvat]** Definiți o clasă numită Rețea. Aceasta va curpinde toate datele rețelei. Rețeaua va fi organizată sub formă de dictionar de noduri (din clasa Nod definită mai sus)

Definiți un constructorul care va primi calea către fisierul de intrare. Se vor citi din fișier datele corespunzătoare rețelei. Se va crea rețeaua Bayesiană având un dicționar cu cheile egale cu identificatorii nodurilor si valorile egale cu obiectele de tip Nod corespunzătoare. Se vor memora nodurile cu valorile observate.

3) **[deja rezolvat]** Implementați în clasa retea metodele definite la curs (toți parametrii sunt noduri; deci variabile aleatorii):

* interogare\_retea(self, nod)
* suport\_except(self, x,v)
* dovezi\_except(self, x,v)

Afișați la fiecare pas calculele intermediare făcute pentru fiecare nod.

4) Se va implementa în clasa Retea o funcție numită valideaza() care să verifice dacă graful direcționat (orientat) dat în fișierul de intrare poate fi o rețea Bayesiană. Veți testa dacă:

* este acicilic
* s-au dat probabilități pentru toate combinațiile posibile de valori booleene ale cauzelor (părinților) pentru un nod dat.
* probabilitățile sunt corecte ( în intervalul [0,1])

5) Definiți în clasa Retea o funcție numită **blocat\_condiționat(drum, multime\_noduri)** care verifică dacă drumul este blocat condiționat de mulțimea de noduri. Atât drumul cât și mulțimea de noduri vor fi date ca liste de identificatori de noduri. Dacă unul dintre identificatorii din liste nu există în rețea, metoda va arunca o eroare.

6) (depinde de realizarea exercitiului 5) Definiți în clasa Retea o funcție numită **d\_separa(x,y, e)** care verifică dacă mulțimea **e** d-separă mulțimile **x** și **y**.

7) Pentru programul implementat la laborator, faceți următoarele îmbunătățiri:

* identificatorii vor avea un cod numeric asociat, de la 0 la N-1, unde N e numărul de noduri din rețeaua Bayesiană.
* se vor folosi doar codurile numerice în listele de identificatori
* parinții și fiii vor fi memorați ca referințe către nodurile corespunzătoare (în loc să fie memorați identificatorii)
* tabelul de probabilități va fi memorat ca o listă de tupluri de forma (cod, probabilitate). codul va fi o sumă de puteri ale lui 2 cu exponent mai mic strict decât numărul de parinți ai nodului. Codul va indica valorile True/False pentru părinți.

8) Pentru programul implementat la laborator să se afișeze pentru o variabilă dată suportul cauzal, E+X (toate variabilele dovezi conectate prin intermediul părinților) și suportul probatoriu, E-X (variabilele dovezi conectate prin intermediul fiilor). Atenție, nu e vorba doar de părinți și fii, ci de **toate nodurile**.

9) **Exercitiu de laborator.**

Relizați următoarele taskuri:

* (1p) Modificati fișierul sezon\_ploios.txt astfel încât să conțină și arcul care lipsește din imagine, împreună cu probabilitățile asociate.
* (1p) Adăugați încă un nod în fișierul input1.txt neschimbând observațiile, astfel încât să creșteți probabilitatea ca studentul să promoveze materia.
* (3p) Definiți propria rețea Bayesiană pentru ce problemă doriți voi, cu următoarele restricții: rețeaua să aibă 5 noduri, rețeaua să fie conexă (nu tare conexă), să nu existe bucle în graful neorientat rezultat din scoaterea sensurilor de pe arce, să aveți minim 2 observații.

10) Definiți propria rețea Bayesiană pentru ce problemă doriți voi, cu următoarele restricții:

* rețeaua să aibă minim 10 noduri
* rețeaua să fie conexă (nu tare conexă) - adică să existe un drum nedirecționat între orica două noduri din rețea.
* să existe:
  + măcar un nod cu minim 3 părinți:
  + măcar un nod cu minim 3 fii
  + minim un nod cu 2 părinți
  + minim un nod cu 2 fii
  + măcar un drum direcționat cu lungime minim 5.

11) Implementați rețeaua Bayesiană din **input1.txt** folosind modulul **pgmpy** (<https://pgmpy.org/>). Veți folosi aceleași variabile observate și aceeași variabilă de interogare din fișier. Implementarea se va face automat pe baza fișierului (parsând fisierul, nu manual!).