



Sisteme Inteligente

Proiect laborator 2019-2020

Adrian Groza, Anca Marginean and Radu Razvan Slavescu
Tool: Pomegranate și AIspace

Name: Roxana Cioarea - Cristescu
Group: 30235
Email: roxana.cioarea@gmail.com

Assoc. Prof. dr. eng. Adrian Groza
Adrian.Groza@cs.utcluj.ro



Contents

1	Prezentare cerință	3
1.1	Descriere narativă	3
1.2	Factori	3
1.3	Specificați	3
2	Rezultate preliminare	4
3	Detali de implementare	6
3.1	Cod relevant	8
4	Aspecte ale proiectului	10
4.1	Avantaje și limitari ale soluției	10
4.2	Posibilități de extindere	10
5	Demo proiect	11
A	Your original code	13

Chapter 1

Prezentare cerință

1.1 Descriere narativă

Proiectul meu este bazat pe structura și specificul Organizației Studenților din Universitatea Tehnică. Avem o rețea Bayesiană care conține informații despre noii recruți și aspecte care le pot afecta acestora activitatea în organizație. Obiectivul nostru este de a calcula care este probabilitatea ca un recrut să devină membru activ (sau oricare dintre celelalte stataturi specifice OSUT)

1.2 Factori

Pentru a putea calcula probabilitatea ca un nou recrut să activeze (și astfel să devină membru activ) am luat în considerare un număr de factori relevanți, aceștia sunt:

1. Anul universitar al recrutului
2. Experiența Big Brother-ului (membru vechi care va avea grija de el)
3. Dacă recrutul este la al doilea interviu (a mai încercat în trecut)

Am considerat important și dacă în cadrul organizației au loc proiecte, cu cât ponderea de proiecte este mai mare cu atât noul recrut trebuie să activeze mai mult pt a deveni membru activ, aici am considerat importante:

1. Vremea
2. Dacă proiectul are loc afară sau înăuntru
3. Dacă este un proiect mare sau mic
4. Dacă are loc o epidemie (pentru că este în temă cu prezentul)

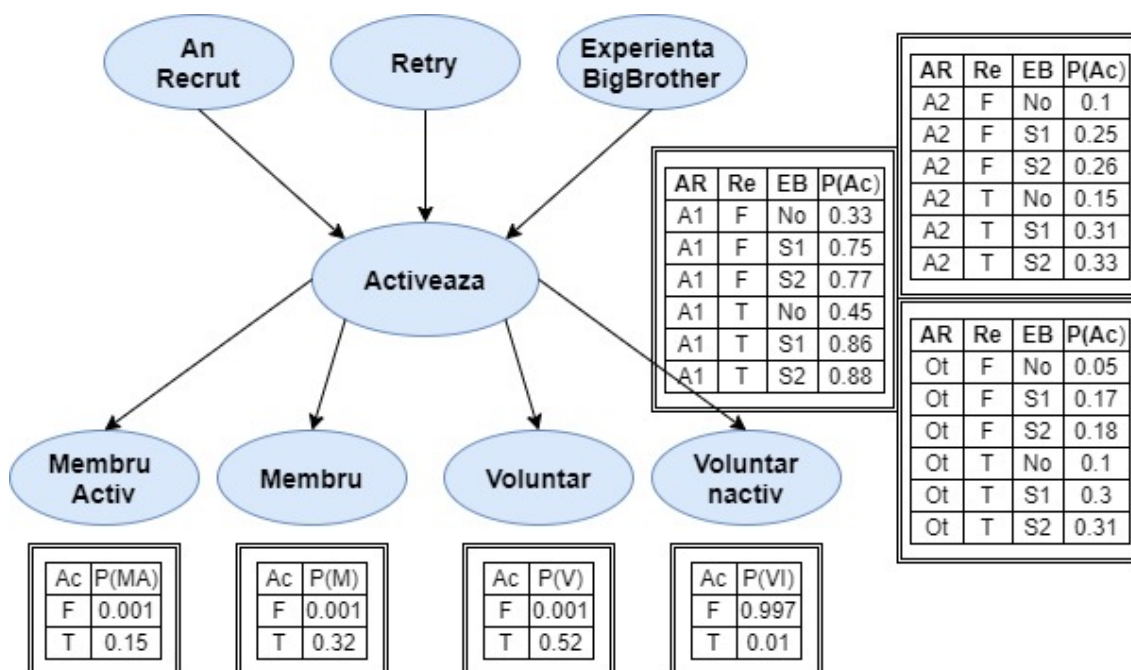
1.3 Specificații

Pentru realizarea acestui proiect voi folosi mediile de proiectare AIspace și Visual Studio, iar codul va fi realizat în Python, folosind librăria Pomegranate.

Chapter 2

Rezultate preliminare

Pe parcursul ultimelor săptămâni am adăugat mai multe aspecte în rețeaua bayesiană. La început aceasta arăta în mare parte astfel:



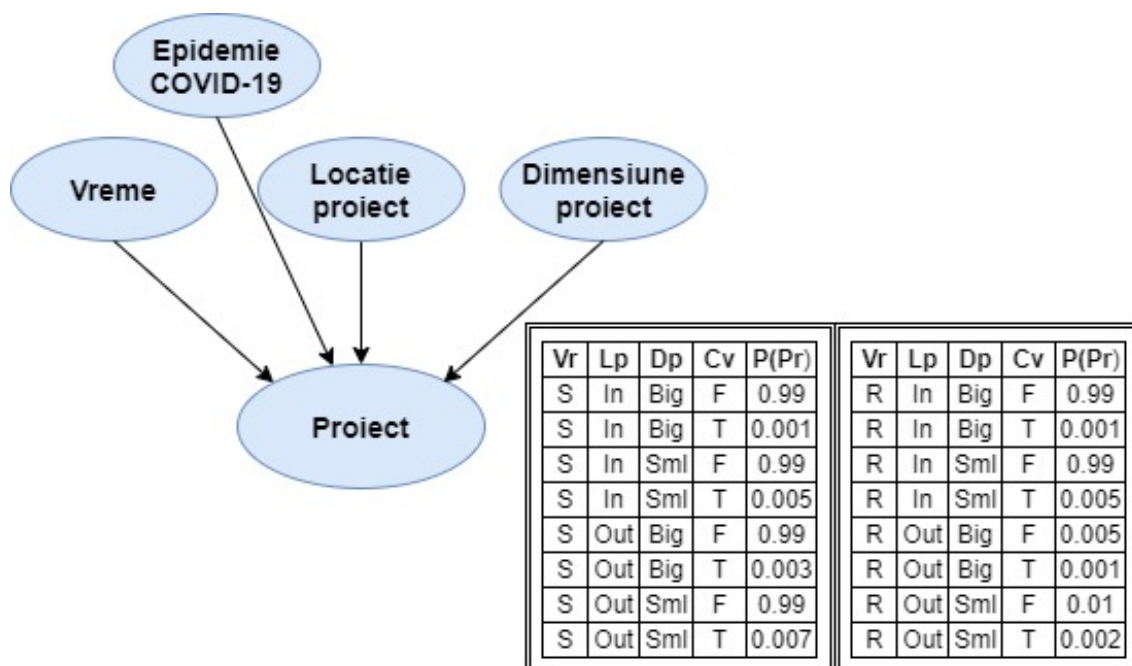
Aici se pot observa factori relevanți pentru activitatea recrutilui, după cum au fost menționați în capitolul anterior:

1. An recrut poate lua valorile A1 (an 1), A2 (an 2), Ot (other)
2. Retry este T (true) sau F (false)
3. Experiență BigBrother poate fi No (nu are), S1 (un semestru), S2 (două semestre)

Statuturile sunt de asemenea separate în acest exemplu, fiecare statut având valorile T (true) și F (false).

Astfel această parte a rețelei ne poate spune care este probabilitatea ca un recrut să activeze, și în funcție de asta care este probabilitatea să devină Membru activ, Membru, Voluntar sau voluntar inactiv.

În următorul pas am adăugat factorii și probabilitățile relevante pentru realizarea unui proiect



Aici deasemenea se pot observa valorile posibile a le intrărilor:

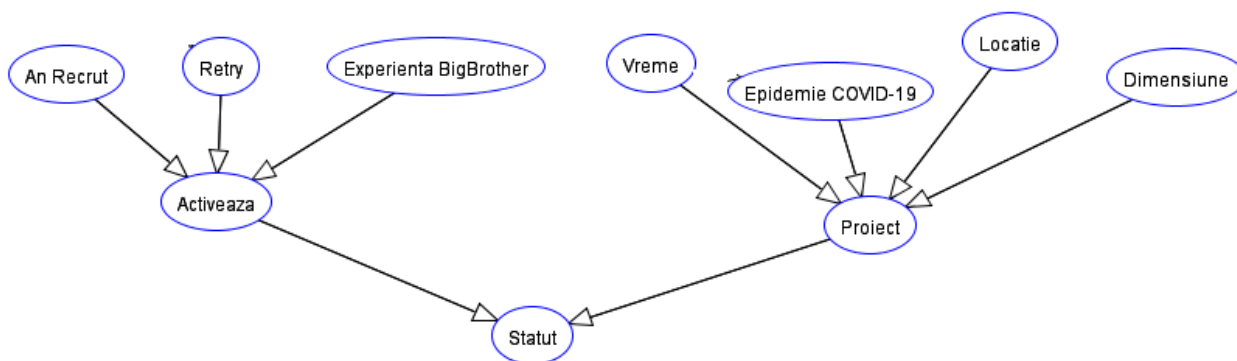
1. Vreme poate lua valorile S (soare) și R (rain/ploaie)
2. Epidemie COVID-19 este T (true) sau F (false)
3. Locație proiect poate fi In (înăuntru) sau Out (afară)
4. Dimensiune proiect se referă ca acesta poate fi Big (mare) sau Sml (mic)

De aici probabilitatea care se poate calcula este de a avea loc un anumit proiect.

Chapter 3

Detali de implementare

Pentru a putea realiza o schemă mai corectă am folosit AIspace, asadar următoarele imagini vor fi ale schemei realizate acolo:



În această schemă se pot observa principalele schimbări pe care le-am realizat de la prima implementare, cele mai relevante fiind:

1. Statutul voluntarilor se găsește într-un singur nor
2. Probabilitatea de retry este condiționată de semestrul în care ne aflăm
3. Probabilitatea epidemiei este afectată de vreme
4. Statutul depinde și de realizarea proiectelor, nu doar de cât de mult activează recruții (deoarece dacă sunt puține proiecte atunci recruți chiar dacă activează mai puțin pot avea statut mai înalt)

În screenshot-urile de mai jos se pot observa probabilitățile pentru fiecare valoare a nodurilor, cat și condiționările (unde este cazul). În cea mai mare parte probabilitățile au fost selectate să se potrivească cu datele din referințele [4] și [5] cât și cunoștințele mele despre organizație și activitatea acesteia

Probability Table for Experienta BigBrother

	$P(\text{Experienta BigBrother}=\text{No})$	$P(\text{Experienta BigBrother}=\text{S1})$	$P(\text{Experienta BigBrother}=\text{S2})$
Prior Probability	0.5	0.28	0.22

No observed value for this node.

OK Cancel

Probability Table for An Recrut

	$P(\text{An Recrut}=\text{A1})$	$P(\text{An Recrut}=\text{A2})$	$P(\text{An Recrut}=\text{A3})$	$P(\text{An Recrut}=\text{A4})$
Prior Probability	0.81	0.158	0.03	0.002

No observed value for this node.

OK Cancel

Probability Table for Semestru

	$P(\text{Semestru}=\text{S1})$	$P(\text{Semestru}=\text{S2})$
Prior Probability	0.5	0.5

No observed value for this node.

OK Cancel

Probability Table for Retry

Semestru	$P(\text{Retry}=\text{T})$	$P(\text{Retry}=\text{F})$
S1	0.001	0.999
S2	0.4	0.6

No observed value for this node.

OK Cancel

Probability Table for Vreme

	$P(\text{Vreme}=\text{S})$	$P(\text{Vreme}=\text{R})$
Prior Probability	0.55	0.45

No observed value for this node.

OK Cancel

Probability Table for Epidemie COVID-19

Vreme	$P(\text{Epidemie COVID-19}=\text{T})$	$P(\text{Epidemie COVID-19}=\text{F})$
S	0.001	0.999
R	0.005	0.995

No observed value for this node.

OK Cancel

Probability Table for Locatie

	$P(\text{Locatie}=\text{In})$	$P(\text{Locatie}=\text{Out})$
Prior Probability	0.5	0.5

No observed value for this node.

OK Cancel

Probability Table for Dimensiune

	$P(\text{Dimensiune}=\text{Big})$	$P(\text{Dimensiune}=\text{Sml})$
Prior Probability	0.3	0.7

No observed value for this node.

OK Cancel

Probability Table for Activeaza

An Recrut	Retry	Experienta BigBrother	$P(\text{Activeaza}=\text{T})$	$P(\text{Activeaza}=\text{F})$
A1	T	No	0.45	0.55
A1	T	S1	0.86	0.14
A1	T	S2	0.88	0.12
A1	F	No	0.33	0.67
A1	F	S1	0.75	0.25
A1	F	S2	0.77	0.23
A2	T	No	0.15	0.85
A2	T	S1	0.31	0.69
A2	T	S2	0.33	0.67
A2	F	No	0.1	0.9
A2	F	S1	0.25	0.75
A2	F	S2	0.26	0.74
A3	T	No	0.1	0.9
A3	T	S1	0.3	0.7
A3	T	S2	0.31	0.69
A3	F	No	0.05	0.95
A3	F	S1	0.17	0.83
A3	F	S2	0.18	0.82
A4	T	No	0.01	0.99
A4	T	S1	0.015	0.985
A4	T	S2	0.018	0.982
A4	F	No	0.001	0.999
A4	F	S1	0.007	0.993
A4	F	S2	0.007	0.993

No observed value for this node.

OK Cancel

Probability Table for Proiect

Vreme	Locatie	Dimensiune	Epidemie COVID-19	$P(\text{Proiect}=\text{T})$	$P(\text{Proiect}=\text{F})$
S	In	Big	T	0.001	0.999
S	In	Big	F	0.99	0.01
S	In	Sml	T	0.005	0.995
S	In	Sml	F	0.99	0.01
S	Out	Big	T	0.003	0.997
S	Out	Big	F	0.99	0.01
S	Out	Sml	T	0.005	0.995
S	Out	Sml	F	0.99	0.01
R	In	Big	T	0.001	0.999
R	In	Big	F	0.99	0.01
R	In	Sml	T	0.005	0.995
R	In	Sml	F	0.99	0.01
R	Out	Big	T	0.001	0.999
R	Out	Big	F	0.005	0.995
R	Out	Sml	T	0.002	0.998
R	Out	Sml	F	0.01	0.99

No observed value for this node.

OK Cancel

Probability Table for Statut					
Proiect	Activeaza	$P(\text{Statut}=MA)$	$P(\text{Statut}=M)$	$P(\text{Statut}=V)$	$P(\text{Statut}=VI)$
T	T	0.15	0.32	0.52	0.01
T	F	0.01	0.01	0.01	0.97
F	T	0.28	0.56	0.15	0.01
F	F	0.02	0.03	0.1	0.85

No observed value for this node.

OK Cancel

3.1 Cod relevant

Un prim pas în realizarea codului este definirea nodurilor, mai jos se pot observa 2 exemple

1. Pentru Distribuții discrete - acestea fiind folosite pentru nodurile a căror valori nu depind de nici un alt nod
2. Pentru probabilități condiționate - acestea fiind folosite pentru nodurile a căror valori depind de unul sau mai multe noduri precedente

```
#Sememstrul din anul Universitar
sem = DiscreteDistribution({'s1': 0.5, 's2': 0.5})

#Recrutul aplica din nou
retry = ConditionalProbabilityTable(
[[ 's1', 'T', 0.001 ],
[ 's1', 'F', 0.999 ],
[ 's2', 'T', 0.4 ],
[ 's2', 'F', 0.6 ]], [sem])
```

După ce toate nodurile au fost definite în acest (pentru codul complet vezi Apendix A), următorul pas este de a crea rețeaua și stările (nodurile) care trebuie mai apoi adăugate în rețea:

```
s11 = State(recrut, name="recrut")
s12 = State(exbig, name="exbig")
s13 = State(sem, name="sem")
s14 = State(retry, name="retry")
s15 = State(activity, name="activity")

activitate = BayesianNetwork("Activitate membri organizatie")
activitate.add_states(s11, s12, s13, s14, s15)
```


Mai apoi adăugăm muchiile, acestea ne spun care noduri sunt părinți căror noduri (primul nod/stare este părintele), cu alte cuvinte acestea sunt liniile care se puteau observa în schemele anterioare. Nu în ultimul rând modelul (adică rețeaua) trebuie ”gătit”.

```
activitate.add_edge(s13, s14)
activitate.add_edge(s11, s15)
activitate.add_edge(s12, s15)
activitate.add_edge(s14, s15)

activitate.bake()
```

Chapter 4

Aspecte ale proiectului

4.1 Avantaje și limitari ale soluției

Prin soluția realizată de mine este posibilă calcularea unor aspecte generale, care deși pot fi corecte în cazul semestrului anterior ar trebui să fie schimbate în funcție de situațiile de la momentul respectiv, cu toate că se poate calcula probabilitatea ca recrutul x să fie activ, rezultatele generale pot varia.

4.2 Posibilități de extindere

Un aspect important al acestui proiect este că nu conține toate informațiile posibile din lumea reală, nici pe departe, așa că mereu vor exista următoarele posibilități de extindere:

1. Adăugarea de noi noduri (acestea ar putea și facultatea, experiența precedentă, motivația la interviu).
2. Mai multe relații dintre noduri, retry este momentan conditionat doar de semestru, care deși este corect pentru recruții de anul 1, nu se aplică și la cei de anul 2, sau 3.
3. Nodurile pot avea mai multe valori (vremea nu este doar ploioasă sau însorită).

Chapter 5

Demo proiect

Pentru a face un demo al proiectului am scris următoarele linii de cod:

```
print(activitate.probability(numpy.array(['A1', 'No',  
|     'S1', 'F', 'T'], ndmin = 2 )))  
print(project.probability(numpy.array(['S', 'Big',  
|     'In', 'F', 'T'], ndmin = 2 )))  
print(status.probability(numpy.array(['A1', 'No',  
|     'S1', 'F', 'T', 'S', 'Big', 'In', 'F', 'T',  
|     'MA'], ndmin = 2 )))
```

Situațiile calculate sunt după cum urmează:

1. Recrutul este în anul 1, bigbrother-ul nu are experiență, ne aflăm în semestrul 1, recrutul nu a mai încercat să intre, și presupunem că activează. Care este probabilitatea ca toate acestea să aibă loc?
2. Este o zi însorită, avem un proiect mare, care are loc înăuntru, epidemia nu s-a întâmplat, și presupunem că proiectul are loc. Care este probabilitatea ca toate acestea să aibă loc?
3. Pentru ultimul exemplu cele 2 cazuri precedente sunt combinate cu faptul ca va ajunge Voluntar inactiv. Care este probabilitatea ca toate acestea să aibă loc?

Mai jos se pot observa rezultatele date de program:

```
PS C:\Users\roxyc\Desktop\Facultate Files\VS work> python Si-proj.py  
0.06675817500000003  
0.081593325  
0.0008170532203772815
```

Pentru a verifica aceste rezultate am făcut înmulțiriile manual, cu alte cuvinte:

$$1. P(\text{activitate}) = P(A1) \cdot P(\text{No}) \cdot P(S1) \cdot P(F/S1) \cdot P(T/A1, \text{No}, F)$$

$$2. P(\text{project}) = P(S) \cdot P(\text{Big}) \cdot P(\text{In}) \cdot P(F/S) \cdot P(T/S, \text{Big}, \text{In}, F)$$

$$3. P(\text{status}) = P(\text{activitate}) \cdot P(\text{project}) \cdot P(\text{MA/activitate, project})$$

În imaginile următoare se pot vedea rezultatele calculate prin înmulțirile menționate anterior și se poate observa că rezultatele coincid.

<small>$0.81 \times 0.5 \times 0.5 \times 0.999 \times 0.33 =$</small> 0.066758175	<small>$0.55 \times 0.3 \times 0.5 \times 0.999 \times 0.99 =$</small> 0.081593325	<small>$0.081593325 \times 0.066758175 \times 0.15 =$</small> 8.170532203772813e-4
---	---	---

Appendix A

Your original code

This section should contain only code developed by you, without any line re-used from other sources.

```
#Anul in care studentul se afla
recrut = DiscreteDistribution({'A1': 0.81, 'A2': 0.158, 'A3': 0.03, 'A4': 0.002})

#Experienta big brother-ului
exbig = DiscreteDistribution({'No': 0.5, 'S1': 0.28, 'S2': 0.22})

#Sememstrul din anul Universitar
sem = DiscreteDistribution({'S1': 0.5, 'S2': 0.5})

#Recrutul aplica din nou
retry = ConditionalProbabilityTable(
    [[ 'S1', 'T', 0.001 ],
     [ 'S1', 'F', 0.999 ],
     [ 'S2', 'T', 0.4 ],
     [ 'S2', 'F', 0.6 ]], [sem])

#Daca recrutul activeaza este dependent de anul in care e si de big-ul pe care il are
activity = ConditionalProbabilityTable(
    [[ 'A1', 'T', 'No', 'T', 0.45 ],
     [ 'A1', 'T', 'S1', 'T', 0.86 ],
     [ 'A1', 'T', 'S2', 'T', 0.88 ],
     [ 'A1', 'F', 'No', 'T', 0.33 ],
     [ 'A1', 'F', 'S1', 'T', 0.75 ],
     [ 'A1', 'F', 'S2', 'T', 0.77 ],
     [ 'A2', 'T', 'No', 'T', 0.15 ],
     [ 'A2', 'T', 'S1', 'T', 0.31 ],
     [ 'A2', 'T', 'S2', 'T', 0.33 ],
     [ 'A2', 'F', 'No', 'T', 0.1 ],
     [ 'A2', 'F', 'S1', 'T', 0.25 ],
     [ 'A2', 'F', 'S2', 'T', 0.26 ],
     [ 'A3', 'T', 'No', 'T', 0.1 ],
     [ 'A3', 'T', 'S1', 'T', 0.3 ],
     [ 'A3', 'T', 'S2', 'T', 0.31 ],
     [ 'A3', 'F', 'No', 'T', 0.05 ],
     [ 'A3', 'F', 'S1', 'T', 0.17 ],
```

```

[ 'A3', 'F', 'S2', 'T', 0.18 ],
[ 'A4', 'T', 'No', 'T', 0.01 ],
[ 'A4', 'T', 'S1', 'T', 0.015 ],
[ 'A4', 'T', 'S2', 'T', 0.018 ],
[ 'A4', 'F', 'No', 'T', 0.001 ],
[ 'A4', 'F', 'S1', 'T', 0.007 ],
[ 'A4', 'F', 'S2', 'T', 0.007 ],
[ 'A1', 'T', 'No', 'F', 0.55 ],
[ 'A1', 'T', 'S1', 'F', 0.14 ],
[ 'A1', 'T', 'S2', 'F', 0.12 ],
[ 'A1', 'F', 'No', 'F', 0.67 ],
[ 'A1', 'F', 'S1', 'F', 0.25 ],
[ 'A1', 'F', 'S2', 'F', 0.23 ],
[ 'A2', 'T', 'No', 'F', 0.85 ],
[ 'A2', 'T', 'S1', 'F', 0.69 ],
[ 'A2', 'T', 'S2', 'F', 0.67 ],
[ 'A2', 'F', 'No', 'F', 0.9 ],
[ 'A2', 'F', 'S1', 'F', 0.75 ],
[ 'A2', 'F', 'S2', 'F', 0.74 ],
[ 'A3', 'T', 'No', 'F', 0.9 ],
[ 'A3', 'T', 'S1', 'F', 0.7 ],
[ 'A3', 'T', 'S2', 'F', 0.69 ],
[ 'A3', 'F', 'No', 'F', 0.95 ],
[ 'A3', 'F', 'S1', 'F', 0.83 ],
[ 'A3', 'F', 'S2', 'F', 0.82 ],
[ 'A4', 'T', 'No', 'F', 0.99 ],
[ 'A4', 'T', 'S1', 'F', 0.985 ],
[ 'A4', 'T', 'S2', 'F', 0.982 ],
[ 'A4', 'F', 'No', 'F', 0.999 ],
[ 'A4', 'F', 'S1', 'F', 0.993 ],
[ 'A4', 'F', 'S2', 'F', 0.993 ]], [recrut, retry, exbig])

```

```
#Vremea de afara
```

```
vreme = DiscreteDistribution({'S': 0.55, 'R': 0.45})
```

```
#Dimensiune proiect
```

```
dimensiune = DiscreteDistribution({'Sml': 0.7, 'Big': 0.3})
```

```
#Locatie proiect
```

```
locatie = DiscreteDistribution({'In': 0.5, 'Out': 0.5})
```

```
#Epidemie coronavirus
```

```
covid = ConditionalProbabilityTable(
    [[ 'S', 'T', 0.001 ],
      [ 'S', 'F', 0.999 ],
      [ 'R', 'T', 0.005 ],
      [ 'R', 'F', 0.995 ]], [vreme])
```

```
#Realizarea proiectului
```

```
proiect = ConditionalProbabilityTable(
```

```

[[ 'S', 'In' , 'Big', 'T', 'T', 0.001 ],
[ 'S', 'In' , 'Big', 'F', 'T', 0.99 ],
[ 'S', 'In' , 'Sml', 'T', 'T', 0.005 ],
[ 'S', 'In' , 'Sml', 'F', 'T', 0.99 ],
[ 'S', 'Out', 'Big', 'T', 'T', 0.003 ],
[ 'S', 'Out', 'Big', 'F', 'T', 0.99 ],
[ 'S', 'Out', 'Sml', 'T', 'T', 0.005 ],
[ 'S', 'Out', 'Sml', 'F', 'T', 0.99 ],
[ 'R', 'In' , 'Big', 'T', 'T', 0.001 ],
[ 'R', 'In' , 'Big', 'F', 'T', 0.99 ],
[ 'R', 'In' , 'Sml', 'T', 'T', 0.005 ],
[ 'R', 'In' , 'Sml', 'F', 'T', 0.99 ],
[ 'R', 'Out', 'Big', 'T', 'T', 0.001 ],
[ 'R', 'Out', 'Big', 'F', 'T', 0.005 ],
[ 'R', 'Out', 'Sml', 'T', 'T', 0.002 ],
[ 'R', 'Out', 'Sml', 'F', 'T', 0.01 ],
[ 'S', 'In' , 'Big', 'T', 'F', 0.999 ],
[ 'S', 'In' , 'Big', 'F', 'F', 0.01 ],
[ 'S', 'In' , 'Sml', 'T', 'F', 0.995 ],
[ 'S', 'In' , 'Sml', 'F', 'F', 0.01 ],
[ 'S', 'Out', 'Big', 'T', 'F', 0.997 ],
[ 'S', 'Out', 'Big', 'F', 'F', 0.01 ],
[ 'S', 'Out', 'Sml', 'T', 'F', 0.995 ],
[ 'S', 'Out', 'Sml', 'F', 'F', 0.01 ],
[ 'R', 'In' , 'Big', 'T', 'F', 0.999 ],
[ 'R', 'In' , 'Big', 'F', 'F', 0.01 ],
[ 'R', 'In' , 'Sml', 'T', 'F', 0.995 ],
[ 'R', 'In' , 'Sml', 'F', 'F', 0.01 ],
[ 'R', 'Out', 'Big', 'T', 'F', 0.999 ],
[ 'R', 'Out', 'Big', 'F', 'F', 0.005 ],
[ 'R', 'Out', 'Sml', 'T', 'F', 0.998 ],
[ 'R', 'Out', 'Sml', 'F', 'F', 0.99 ]], [vreme, locatie, dimensiune, covid])

```

#Statut voluntari

```
statut = ConditionalProbabilityTable(
```

```

[[ 'T', 'T', 'MA', 0.15 ],
[ 'T', 'F', 'MA', 0.01 ],
[ 'F', 'T', 'MA', 0.28 ],
[ 'F', 'F', 'MA', 0.02 ],
[ 'T', 'T', 'M' , 0.32 ],
[ 'T', 'F', 'M' , 0.01 ],
[ 'F', 'T', 'M' , 0.56 ],
[ 'F', 'F', 'M' , 0.03 ],
[ 'T', 'T', 'V' , 0.52 ],
[ 'T', 'F', 'V' , 0.01 ],
[ 'F', 'T', 'V' , 0.15 ],
[ 'F', 'F', 'V' , 0.01 ],
[ 'T', 'T', 'VI', 0.01 ],
[ 'T', 'F', 'VI', 0.97 ],
[ 'F', 'T', 'VI', 0.01 ],

```

```

[ 'F', 'F', 'VI', 0.85 ]], [proiect, activity])

s11 = State(recrut, name="recrut")
s12 = State(exbig, name="exbig")
s13 = State(sem, name="sem")
s14 = State(retry, name="retry")
s15 = State(activity, name="activity")

s21 = State(vreme, name="vreme")
s22 = State(dimensiune, name="dimensiune")
s23 = State(locatie, name="locatie")
s24 = State(covid, name="covid")
s25 = State(proiect, name="proiect")

s3 = State(statut, name="statut")

activitate = BayesianNetwork("Activitate membri organizatie")
activitate.add_states(s11, s12, s13, s14, s15)

activitate.add_edge(s13, s14)
activitate.add_edge(s11, s15)
activitate.add_edge(s12, s15)
activitate.add_edge(s14, s15)

project = BayesianNetwork("Proiecte organizatie")
project.add_states(s21, s22, s23, s24, s25)

project.add_edge(s21, s24)
project.add_edge(s21, s25)
project.add_edge(s22, s25)
project.add_edge(s23, s25)
project.add_edge(s24, s25)

status = BayesianNetwork("Statut membri organizatie")
status.add_states(s11, s12, s13, s14, s15, s21, s22, s23, s24, s25, s3)

status.add_edge(s13, s14)
status.add_edge(s11, s15)
status.add_edge(s12, s15)
status.add_edge(s14, s15)

status.add_edge(s21, s24)
status.add_edge(s21, s25)
status.add_edge(s22, s25)
status.add_edge(s23, s25)
status.add_edge(s24, s25)

status.add_edge(s15, s3)
status.add_edge(s25, s3)

```



```
activitate.bake()
project.bake()
status.bake()

print(activitate.probability(numpy.array(['A1','No',
'S1', 'T', 'T'], ndmin = 2 )))
print(project.probability(numpy.array(['S', 'Big',
'In', 'T', 'F', 'VI'], ndmin = 2 )))
print(status.probability(numpy.array(['A1','No',
'S1', 'T', 'T', 'S', 'Big', 'In', 'T', 'F',
'VI'], ndmin = 2 )))
```

Bibliography

[1] Documentația Pomegranate

<https://pomegranate.readthedocs.io/en/docs/BayesianNetwork.html>

[2] Problema Monty Hall + explicații

https://github.com/jmschrei/pomegranate/blob/master/tutorials/B_Model_Tutorial_4_Bayesian_Networks.ipynb

[3] AIspace

<http://www.aispace.org/bayes/index.shtml>

[4] Spreadsheet statut membri OSUT

https://docs.google.com/spreadsheets/d/18tjFWmPxMJlZvg4CxRk0JPTpP3t_Dw_HqF2VBMPCyVc/edit#gid=2085008750

[5] Spreadsheet recruți OSUT

<https://docs.google.com/spreadsheets/d/1bL3c8-LMtASicKNV7mpZSjyGaTgr6xTd7sspvbE1Nhk/edit?fbclid=IwAR2Rv0zPlCBRl7TMZ9H08CZqqvc-8QHik2aswLyFFuLiBkm6kkEDnYYoUly#gid=219098936>

Intelligent Systems Group

