School of Data and Computer Science, Sun Yat-sen University

E09 Bayesian Network

Suixin Ou Linkai Yang

October 26, 2020

Contents

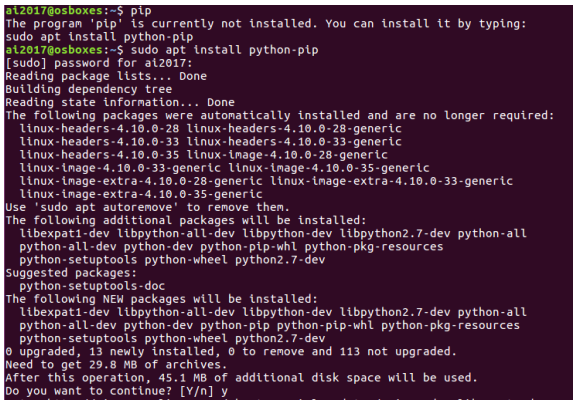
1 Pomegranate Installation

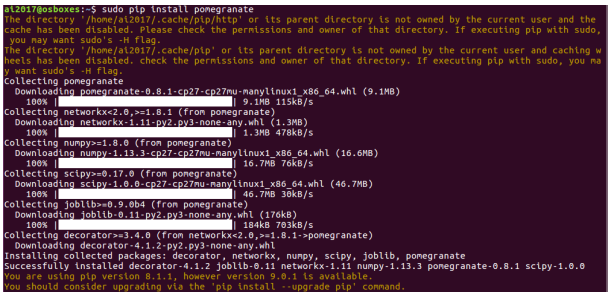
Under Linux:

1. Install `python` first (**python 2**, not python 3).

2. Run `sudo apt-get install python-pip` to install `pip`.

3. Run `sudo pip install pomegranate` to install `pomegranate`.

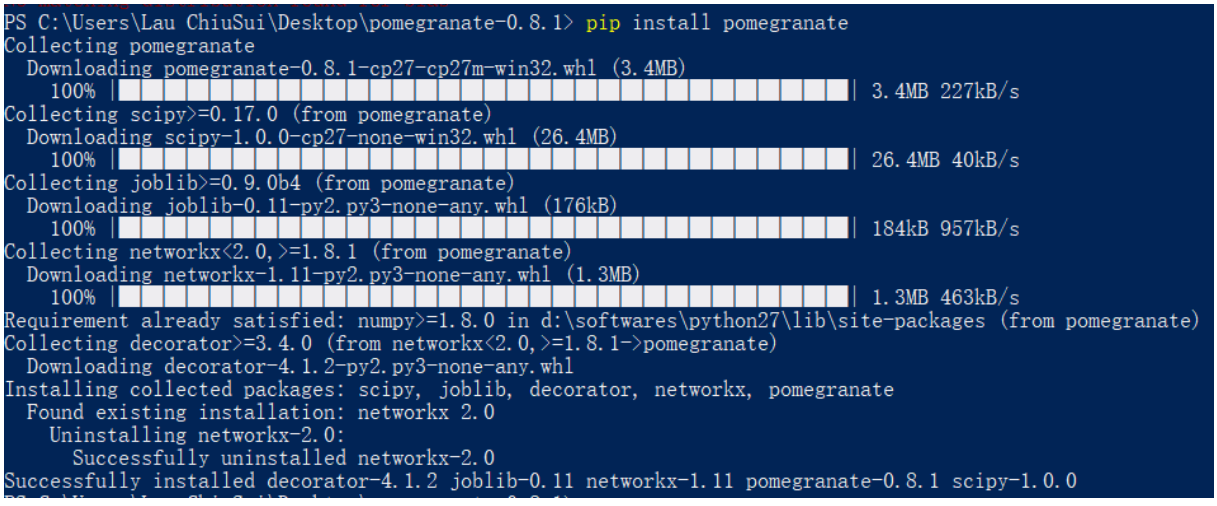




Under Windows

You can also run `pip install pomegranate` if you have installed `pip`. If you don't know how to install `pip`, please click `https://jingyan.baidu.com/article/e73e26c0d94e0524adb6a7ff.html`.

For more, please click the homepage of Pomegranate - `https://github.com/jmschrei/pomegranate` for help.



2 Building Bayesian Network

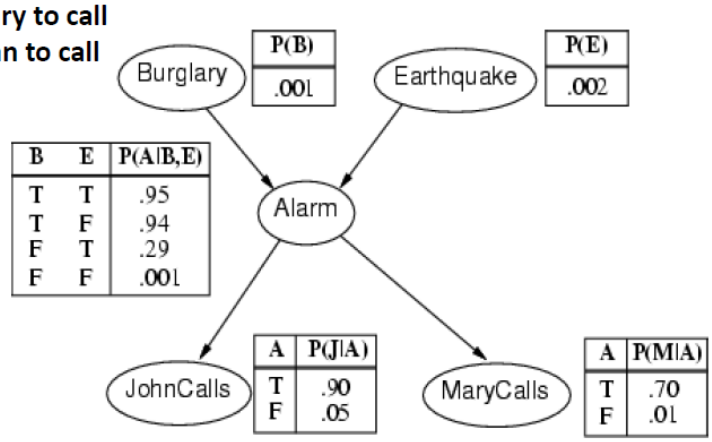
Please refer to `Tutorial_4_Bayesian_Networks.pdf`. I will explain it in class.

3 Tasks

3.1 Burglary

- **A burglary can set the alarm off**
- **An earthquake can set the alarm off**
- **The alarm can cause Mary to call**
- **The alarm can cause John to call**

Note that these tables only provide the probability that Xi is true.
(E.g., Pr(A is true|B,E))
The probability that Xi is false is 1- these values



Please code to calculate:

1. $P(A)$
2. $P(J\overline{M})$
3. $P(A|J\overline{M})$
4. $P(B|A)$
5. $P(B|J\overline{M})$
6. $P(J\overline{M}|\overline{B})$

```
P(Alarm) =
0.002516442

P(J&&~M) =
0.050054875461

P(A |J&&~M) =
0.0135738893313

P(B |A) =
0.373551228282

P(B |J&&~M) =
0.0051298581334

P(J&&~M |~B) =
0.049847949
```

3.2 Diagnosing

Variables and their domais

(1) PatientAge: ['0-30', '31-65', '65+']

(2) CTScanResult: ['Ischemic Stroke', 'Hemmorraghic Stroke']

(3) MRIScanResult: ['Ischemic Stroke', 'Hemmorraghic Stroke']

(4) StrokeType: ['Ischemic Stroke', 'Hemmorraghic Stroke', 'Stroke Mimic']

(5) Anticoagulants: ['Used', 'Not used']

(6) Mortality: ['True', 'False']

(7) Disability: ['Negligible', 'Moderate', 'Severe']

CPTs

Note: [CTScanResult, MRIScanResult,StrokeType] means:

P(StrokeType='...' | CTScanResult='...' $\wedge$ MRIScanResult='...')

(1)

[PatientAge]

['0-30',0.10],

['31-65',0.30],

['65+',0.60]

(2)

[CTScanResult]

['Ischemic Stroke',0.7],

['Hemmorraghic Stroke',0.3]

(3)

[MRIScanResult]

['Ischemic Stroke',0.7],

['Hemmorraghic Stroke',0.3]

(4)

```
[Anticoagulants]

['Used',0.5],
['Not used',0.5]

(5)
[CTScanResult,MRIScanResult,StrokeType]

['Ischemic Stroke','Ischemic Stroke','Ischemic Stroke',0.8],
['Ischemic Stroke','Hemmorraghic Stroke','Ischemic Stroke',0.5],
['Hemmorraghic Stroke','Ischemic Stroke','Ischemic Stroke',0.5],
['Hemmorraghic Stroke','Hemmorraghic Stroke','Ischemic Stroke',0],

['Ischemic Stroke','Ischemic Stroke','Hemmorraghic Stroke',0],
['Ischemic Stroke','Hemmorraghic Stroke','Hemmorraghic Stroke',0.4],
['Hemmorraghic Stroke','Ischemic Stroke','Hemmorraghic Stroke',0.4],
['Hemmorraghic Stroke','Hemmorraghic Stroke','Hemmorraghic Stroke',0.9],

['Ischemic Stroke','Ischemic Stroke','Stroke Mimic',0.2],
['Ischemic Stroke','Hemmorraghic Stroke','Stroke Mimic',0.1],
['Hemmorraghic Stroke','Ischemic Stroke','Stroke Mimic',0.1],
['Hemmorraghic Stroke','Hemmorraghic Stroke','Stroke Mimic',0.1],

(6)
[StrokeType, Anticoagulants, Mortality]

['Ischemic Stroke', 'Used', 'False',0.28],
['Hemmorraghic Stroke', 'Used', 'False',0.99],
['Stroke Mimic', 'Used', 'False',0.1],
['Ischemic Stroke','Not used', 'False',0.56],
['Hemmorraghic Stroke', 'Not used', 'False',0.58],
['Stroke Mimic', 'Not used', 'False',0.05],
```

```
['Ischemic Stroke',  'Used' ,'True',0.72],
['Hemmorraghic Stroke', 'Used', 'True',0.01],
['Stroke Mimic', 'Used', 'True',0.9],
['Ischemic Stroke',  'Not used' ,'True',0.44],
['Hemmorraghic Stroke', 'Not used', 'True',0.42 ],
['Stroke Mimic', 'Not used', 'True',0.95]
```

(7)

```
[StrokeType, PatientAge, Disability]

['Ischemic Stroke',   '0-30','Negligible', 0.80],
['Hemmorraghic Stroke', '0-30','Negligible', 0.70],
['Stroke Mimic',        '0-30', 'Negligible',0.9],
['Ischemic Stroke',     '31-65','Negligible', 0.60],
['Hemmorraghic Stroke', '31-65','Negligible', 0.50],
['Stroke Mimic',        '31-65', 'Negligible',0.4],
['Ischemic Stroke',     '65+'  , 'Negligible',0.30],
['Hemmorraghic Stroke', '65+'  , 'Negligible',0.20],
['Stroke Mimic',        '65+'  , 'Negligible',0.1],

['Ischemic Stroke',     '0-30' ,'Moderate',0.1],
['Hemmorraghic Stroke', '0-30' ,'Moderate',0.2],
['Stroke Mimic',        '0-30' ,'Moderate',0.05],
['Ischemic Stroke',     '31-65','Moderate',0.3],
['Hemmorraghic Stroke', '31-65','Moderate',0.4],
['Stroke Mimic',        '31-65','Moderate',0.3],
['Ischemic Stroke',     '65+'  ,'Moderate',0.4],
['Hemmorraghic Stroke', '65+'  ,'Moderate',0.2],
['Stroke Mimic',        '65+'  ,'Moderate',0.1],

['Ischemic Stroke',     '0-30' ,'Severe',0.1],
['Hemmorraghic Stroke', '0-30' ,'Severe',0.1],
['Stroke Mimic',        '0-30' ,'Severe',0.05],
```

```
['Ischemic Stroke',      '31-65','Severe',0.1],
['Hemmorraghic Stroke', '31-65','Severe',0.1],
['Stroke Mimic',         '31-65','Severe',0.3],
['Ischemic Stroke',      '65+'  ,'Severe',0.3],
['Hemmorraghic Stroke', '65+'  ,'Severe',0.6],
['Stroke Mimic',         '65+'  ,'Severe',0.8]
```

Calculation

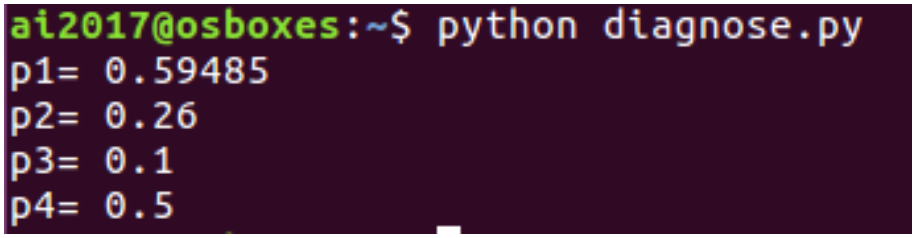
Please code to calculate the following probability value:

p1 = P(Mortality='True' | PatientAge='31-65' ∧ CTScanResult='Ischemic Stroke')

p2 = P(Disability='Moderate' | PatientAge='65+' ∧ MRIScanResult='Hemmorraghic Stroke')

p3 = P(StrokeType='Stroke Mimic' | PatientAge='65+' ∧ CTScanResult='Hemmorraghic Stroke' ∧ MRIScanResult='Ischemic Stroke')

p4 = P(Anticoagulants='Not used' | PatientAge='0-30')



Please solve the 2 tasks and hand in a file named E08_YourNumber.pdf, and send it to ai_2020@foxmail.com

4 Codes and Results

4.1 Codes

4.1.1 Burglary

burglary.py

```
1 from pomegranate import *
2 B = DiscreteDistribution({True: 0.001, False: 0.999})
3 E = DiscreteDistribution({True: 0.002, False: 0.998})
4 A = ConditionalProbabilityTable(
5       [[True, True, True, 0.95],
6        [True, True, False, 0.05],
7        [True, False, True, 0.94],
```

```
8         [True, False, False, 0.06],
9         [False, True, True, 0.29],
10        [False, True, False, 0.71],
11        [False, False, True, 0.001],
12        [False, False, False, 0.999]],
13       [B, E])
14 J = ConditionalProbabilityTable(
15       [[True, True, 0.9],
16        [True, False, 0.1],
17        [False, True, 0.05],
18        [False, False, 0.95]],
19       [A])
20 M = ConditionalProbabilityTable(
21       [[True, True, 0.7],
22        [True, False, 0.3],
23        [False, True, 0.01],
24        [False, False, 0.99]],
25       [A])
26 s0 = State(B, name='B')
27 s1 = State(E, name='E')
28 s2 = State(A, name='A')
29 s3 = State(J, name='J')
30 s4 = State(M, name='M')
31 model = BayesianNetwork('Burglary')
32 model.add_states(s0, s1, s2, s3, s4)
33 model.add_transition(s0, s2)
34 model.add_transition(s1, s2)
35 model.add_transition(s2, s3)
36 model.add_transition(s2, s4)
37 model.bake()
38 PA1 = model.predict_proba({})[2].parameters[0][True]
39 PJ1M0 = model.predict_proba({'M': False})[
40       3].parameters[0][True] * model.predict_proba({})[4].parameters[0][False]
```

```
PA1_J1M0 = model.predict_proba({'J': True, 'M': False})[2].parameters[0][True]
PB1_A1 = model.predict_proba({'A': True})[0].parameters[0][True]
PB1_J1M0 = model.predict_proba({'J': True, 'M': False})[0].parameters[0][True]
PJ1M0_B0 = (1 - PB1_J1M0) * PJ1M0 / \
    model.predict_proba({})[0].parameters[0][False]
print(PA1)
print(PJ1M0)
print(PA1_J1M0)
print(PB1_A1)
print(PB1_J1M0)
print(PJ1M0_B0)
```

4.1.2 Diagnosing

diagnosing.py

```
from pomegranate import *
PatientAge = DiscreteDistribution({'0-30': 0.10, '31-65': 0.30, '65+': 0.60})
CTScanResult = DiscreteDistribution(
    {'Ischemic Stroke': 0.7, 'Hemmorraghic Stroke': 0.3})
MRIScanResult = DiscreteDistribution(
    {'Ischemic Stroke': 0.7, 'Hemmorraghic Stroke': 0.3})
Anticoagulants = DiscreteDistribution({'Used': 0.5, 'Not used': 0.5})
StrokeType = ConditionalProbabilityTable(
    [['Ischemic Stroke', 'Ischemic Stroke', 'Ischemic Stroke', 0.8],
     ['Ischemic Stroke', 'Hemmorraghic Stroke', 'Ischemic Stroke', 0.5],
     ['Hemmorraghic Stroke', 'Ischemic Stroke', 'Ischemic Stroke', 0.5],
     ['Hemmorraghic Stroke', 'Hemmorraghic Stroke', 'Ischemic Stroke', 0],
     ['Ischemic Stroke', 'Ischemic Stroke', 'Hemmorraghic Stroke', 0],
     ['Ischemic Stroke', 'Hemmorraghic Stroke', 'Hemmorraghic Stroke', 0.4],
     ['Hemmorraghic Stroke', 'Ischemic Stroke', 'Hemmorraghic Stroke', 0.4],
     ['Hemmorraghic Stroke', 'Hemmorraghic Stroke', 'Hemmorraghic Stroke', 0.9],
     ['Ischemic Stroke', 'Ischemic Stroke', 'Stroke Mimic', 0.2],
     ['Ischemic Stroke', 'Hemmorraghic Stroke', 'Stroke Mimic', 0.1],
```

```
19        ['Hemmorraghic Stroke', 'Ischemic Stroke', 'Stroke Mimic', 0.1],
20        ['Hemmorraghic Stroke', 'Hemmorraghic Stroke', 'Stroke Mimic', 0.1]],
21      [CTScanResult, MRIScanResult])
22  Mortality = ConditionalProbabilityTable(
23      [['Ischemic Stroke', 'Used', 'False', 0.28],
24       ['Hemmorraghic Stroke', 'Used', 'False', 0.99],
25          ['Stroke Mimic', 'Used', 'False', 0.1],
26          ['Ischemic Stroke', 'Not used', 'False', 0.56],
27          ['Hemmorraghic Stroke', 'Not used', 'False', 0.58],
28          ['Stroke Mimic', 'Not used', 'False', 0.05],
29          ['Ischemic Stroke',  'Used', 'True', 0.72],
30          ['Hemmorraghic Stroke', 'Used', 'True', 0.01],
31          ['Stroke Mimic', 'Used', 'True', 0.9],
32          ['Ischemic Stroke',  'Not used', 'True', 0.44],
33          ['Hemmorraghic Stroke', 'Not used', 'True', 0.42],
34          ['Stroke Mimic', 'Not used', 'True', 0.95]],
35      [StrokeType, Anticoagulants])
36  Disability = ConditionalProbabilityTable(
37      [['Ischemic Stroke',    '0-30', 'Negligible', 0.80],
38       ['Hemmorraghic Stroke', '0-30', 'Negligible', 0.70],
39          ['Stroke Mimic',         '0-30', 'Negligible', 0.9],
40          ['Ischemic Stroke',      '31-65', 'Negligible', 0.60],
41          ['Hemmorraghic Stroke', '31-65', 'Negligible', 0.50],
42          ['Stroke Mimic',         '31-65', 'Negligible', 0.4],
43          ['Ischemic Stroke',      '65+', 'Negligible', 0.30],
44          ['Hemmorraghic Stroke', '65+', 'Negligible', 0.20],
45          ['Stroke Mimic',         '65+', 'Negligible', 0.1],
46          ['Ischemic Stroke',      '0-30', 'Moderate', 0.1],
47          ['Hemmorraghic Stroke', '0-30', 'Moderate', 0.2],
48          ['Stroke Mimic',         '0-30', 'Moderate', 0.05],
49          ['Ischemic Stroke',      '31-65', 'Moderate', 0.3],
50          ['Hemmorraghic Stroke', '31-65', 'Moderate', 0.4],
51          ['Stroke Mimic',         '31-65', 'Moderate', 0.3],
```

```
52          ['Ischemic Stroke',      '65+', 'Moderate', 0.4],
53          ['Hemmorraghic Stroke', '65+', 'Moderate', 0.2],
54          ['Stroke Mimic',         '65+', 'Moderate', 0.1],
55          ['Ischemic Stroke',      '0-30', 'Severe', 0.1],
56          ['Hemmorraghic Stroke', '0-30', 'Severe', 0.1],
57          ['Stroke Mimic',         '0-30', 'Severe', 0.05],
58          ['Ischemic Stroke',      '31-65', 'Severe', 0.1],
59          ['Hemmorraghic Stroke', '31-65', 'Severe', 0.1],
60          ['Stroke Mimic',         '31-65', 'Severe', 0.3],
61          ['Ischemic Stroke',      '65+', 'Severe', 0.3],
62          ['Hemmorraghic Stroke', '65+', 'Severe', 0.6],
63          ['Stroke Mimic',         '65+', 'Severe', 0.8]],
64      [StrokeType, PatientAge])
65  s0 = State(PatientAge, name='PatientAge')
66  s1 = State(CTScanResult, name='CTScanResult')
67  s2 = State(MRIScanResult, name='MRIScanResult')
68  s3 = State(Anticoagulants, name='Anticoagulants')
69  s4 = State(StrokeType, name='StrokeType')
70  s5 = State(Mortality, name='Mortality')
71  s6 = State(Disability, name='Disability')
72  model = BayesianNetwork('Diagnosing')
73  model.add_states(s0, s1, s2, s3, s4, s5, s6)
74  model.add_transition(s1, s4)
75  model.add_transition(s2, s4)
76  model.add_transition(s4, s5)
77  model.add_transition(s3, s5)
78  model.add_transition(s4, s6)
79  model.add_transition(s0, s6)
80  model.bake()
81  p1 = model.predict_proba(
82      {'PatientAge': '31-65', 'CTScanResult': 'Ischemic Stroke'})\
83          [5].parameters[0]['True']
84  p2 = model.predict_proba(
```

```
85      {'PatientAge': '65+', 'MRIScanResult': 'Hemmorraghic Stroke'})\
86          [6].parameters[0]['Moderate']
87  p3 = model.predict_proba(
88      {'PatientAge': '65+', 'CTScanResult': 'Hemmorraghic Stroke',
89                            'MRIScanResult': 'Ischemic Stroke'})\
90                                [4].parameters[0]['Stroke Mimic']
91  p4 = model.predict_proba({'PatientAge': '0-30'})\
92      [3].parameters[0]['Not used']
93  print(p1)
94  print(p2)
95  print(p3)
96  print(p4)
```

4.2 Results

4.2.1 Burglary

```
> python .\burglary.py
0.0025164420000009344
0.05005487546100036
0.013573889331311458
0.37355122828189946
0.005129858133403523
0.04984794900000027
```

Figure 1: Result of burglary.py

4.2.2 Diagnosing

```
> python .\burglary.py
0.0025164420000009344
0.05005487546100036
0.013573889331311458
0.37355122828189946
0.005129858133403523
0.04984794900000027
```

Figure 2: Result of diagnosing.py