Relation for FAIKR Module 3 Project

Daniele Baiocco, matr. 0001057671

April 4, 2022

Contents

1 Lung cancer dataset

The dataset I've used is public available at this link. It has 16 columns, I've used 10 of them in order to build the Bayesian Network:

the **age** of the patient, whether he/she suffers from **anxiety**, whether he/she **coughs**, **wheezes**, **smokes**, whether he/she is **allergic** to something or not, whether he/she **fatigues** easily or not, whether he/she has **lung cancer**, **chest pain** and whether there was a **peer pressure** to smoke when he/she was younger.

All the code, including this presentation and the dataset, are available at this github repository, made by me.

2 Preprocessing on the dataset

After importing the dataset with

```
pd.read_csv('survey_lung_cancer.csv')
```

I've done the following things:

- I dropped the columns **'GENDER','YELLOW_FINGERS','CHRONIC DISEASE','ALCOHOL CONSUMING','SWALLOWING DIFFICULTY','SHORTNESS OF BREATH'**.
- I discretized the attribute **'AGE'** in three bins by using **KBinsDiscretizer** from **sklearn.preprocessing**.
- For each column that had the boolean values true and false encoded respectively as 2 and 1, I changed the encoding in **'YES'** and **'NO'** to make things clearer:

```
for i in df1.iloc[:, 1:9]:
df1[i] = df1[i].transform(lambda x: 'YES'
                          if (x == 2) else 'NO')
```

3 Building the Bayesian Network

After quite a while, I ended up with the Bayesian Network shown in Figure 1. I built it following a **causal reasoning** because by doing so usually we end up having to specify fewer numbers, and the numbers will often be easier to come up with. I created this network by instantiating the BayesianNetwork class from pgmpy.models with an array of tuples in which each tuple represent an arc between two nodes. I then used the fit method of the BayesianNetwork, passing as arguments the dataset along with the MaximumLikelihoodEstimator from pgmpy.estimators, in order to compute the **conditional probability distribution**:

```
model= BayesianNetwork(nodes)
model.fit(df1, estimator=MaximumLikelihoodEstimator)
```

The computed tables are shown in Figure 2.

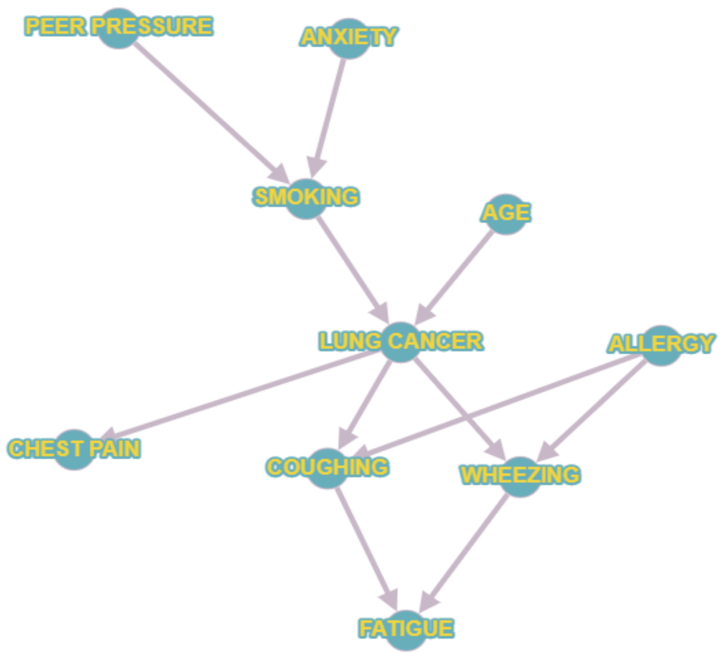


Figure 1: Bayesian Network built from the attributes in the lung cancer dataset

4 Inference

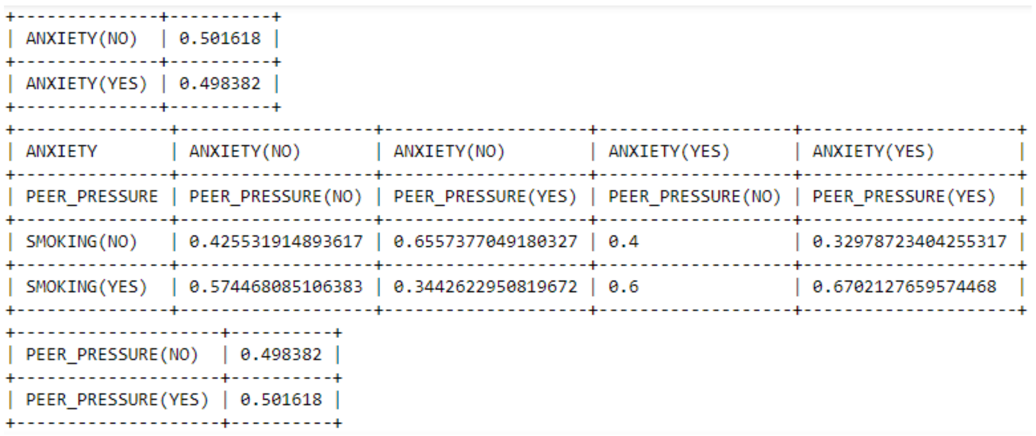
I wrapped the BayesianNetwork instance inside a VariableElimination instance. The class is taken from pgmpy.inference and It gives a way to compute queries through the **query** method:

```
from pgmpy.inference import VariableElimination
lungcancer_infer = VariableElimination(model)
```

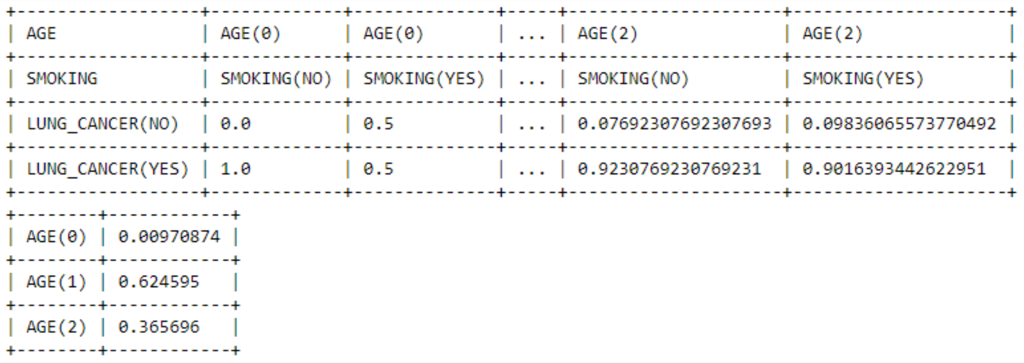
4.1 Computing variable elimination

I did the following query:

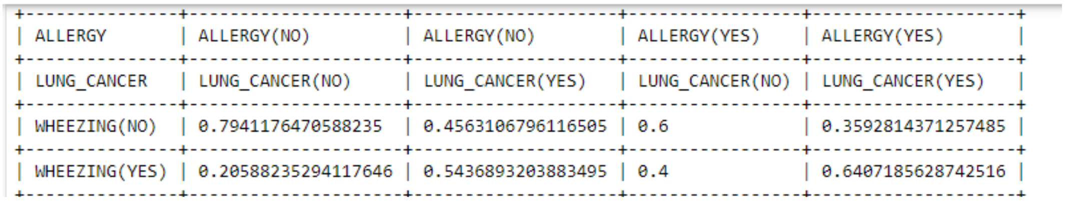
```
q1=lungcancer_infer.query(variables=['LUNG_CANCER'],
                          evidence={'PEER_PRESSURE':'YES'})
```



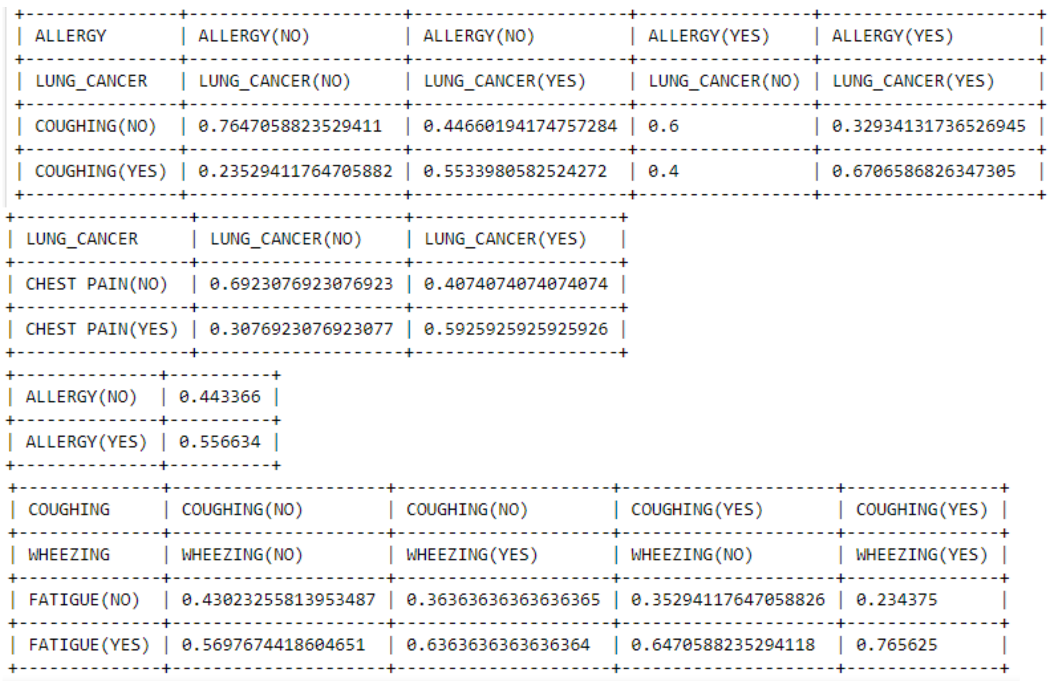
```
+--------------+----------+
| ANXIETY(NO)  | 0.501618 |
+--------------+----------+
| ANXIETY(YES) | 0.498382 |
+--------------+----------+
+---------------+--------------------+--------------------+-------------------+---------------------+
| ANXIETY       | ANXIETY(NO)        | ANXIETY(NO)        | ANXIETY(YES)      | ANXIETY(YES)        |
+---------------+--------------------+--------------------+-------------------+---------------------+
| PEER_PRESSURE | PEER_PRESSURE(NO)  | PEER_PRESSURE(YES) | PEER_PRESSURE(NO) | PEER_PRESSURE(YES)  |
+---------------+--------------------+--------------------+-------------------+---------------------+
| SMOKING(NO)   | 0.425531914893617  | 0.6557377049180327 | 0.4               | 0.32978723404255317 |
+---------------+--------------------+--------------------+-------------------+---------------------+
| SMOKING(YES)  | 0.574468085106383  | 0.3442622950819672 | 0.6               | 0.6702127659574468  |
+---------------+--------------------+--------------------+-------------------+---------------------+
+--------------------+----------+
| PEER_PRESSURE(NO)  | 0.498382 |
+--------------------+----------+
| PEER_PRESSURE(YES) | 0.501618 |
+--------------------+----------+
```



```
+------------------+-------------+--------------+-----+---------------------+---------------------+
| AGE              | AGE(0)      | AGE(0)       | ... | AGE(2)              | AGE(2)              |
+------------------+-------------+--------------+-----+---------------------+---------------------+
| SMOKING          | SMOKING(NO) | SMOKING(YES) | ... | SMOKING(NO)         | SMOKING(YES)        |
+------------------+-------------+--------------+-----+---------------------+---------------------+
| LUNG_CANCER(NO)  | 0.0         | 0.5          | ... | 0.07692307692307693 | 0.09836065573770492 |
+------------------+-------------+--------------+-----+---------------------+---------------------+
| LUNG_CANCER(YES) | 1.0         | 0.5          | ... | 0.9230769230769231  | 0.9016393442622951  |
+------------------+-------------+--------------+-----+---------------------+---------------------+
+--------+------------+
| AGE(0) | 0.00970874 |
+--------+------------+
| AGE(1) | 0.624595   |
+--------+------------+
| AGE(2) | 0.365696   |
+--------+------------+
```



```
+---------------+---------------------+--------------------+-----------------+--------------------+
| ALLERGY       | ALLERGY(NO)         | ALLERGY(NO)        | ALLERGY(YES)    | ALLERGY(YES)       |
+---------------+---------------------+--------------------+-----------------+--------------------+
| LUNG_CANCER   | LUNG_CANCER(NO)     | LUNG_CANCER(YES)   | LUNG_CANCER(NO) | LUNG_CANCER(YES)   |
+---------------+---------------------+--------------------+-----------------+--------------------+
| WHEEZING(NO)  | 0.7941176470588235  | 0.4563106796116505 | 0.6             | 0.3592814371257485 |
+---------------+---------------------+--------------------+-----------------+--------------------+
| WHEEZING(YES) | 0.20588235294117646 | 0.5436893203883495 | 0.4             | 0.6407185628742516 |
+---------------+---------------------+--------------------+-----------------+--------------------+
```



```
+---------------+---------------------+---------------------+-----------------+---------------------+
| ALLERGY       | ALLERGY(NO)         | ALLERGY(NO)         | ALLERGY(YES)    | ALLERGY(YES)        |
+---------------+---------------------+---------------------+-----------------+---------------------+
| LUNG_CANCER   | LUNG_CANCER(NO)     | LUNG_CANCER(YES)    | LUNG_CANCER(NO) | LUNG_CANCER(YES)    |
+---------------+---------------------+---------------------+-----------------+---------------------+
| COUGHING(NO)  | 0.7647058823529411  | 0.44660194174757284 | 0.6             | 0.32934131736526945 |
+---------------+---------------------+---------------------+-----------------+---------------------+
| COUGHING(YES) | 0.23529411764705882 | 0.5533980582524272  | 0.4             | 0.6706586826347305  |
+---------------+---------------------+---------------------+-----------------+---------------------+
+-----------------+--------------------+--------------------+
| LUNG_CANCER     | LUNG_CANCER(NO)    | LUNG_CANCER(YES)   |
+-----------------+--------------------+--------------------+
| CHEST PAIN(NO)  | 0.6923076923076923 | 0.4074074074074074 |
+-----------------+--------------------+--------------------+
| CHEST PAIN(YES) | 0.3076923076923077 | 0.5925925925925926 |
+-----------------+--------------------+--------------------+
+--------------+----------+
| ALLERGY(NO)  | 0.443366 |
+--------------+----------+
| ALLERGY(YES) | 0.556634 |
+--------------+----------+
+--------------+---------------------+---------------------+---------------------+---------------+
| COUGHING     | COUGHING(NO)        | COUGHING(NO)        | COUGHING(YES)       | COUGHING(YES) |
+--------------+---------------------+---------------------+---------------------+---------------+
| WHEEZING     | WHEEZING(NO)        | WHEEZING(YES)       | WHEEZING(NO)        | WHEEZING(YES) |
+--------------+---------------------+---------------------+---------------------+---------------+
| FATIGUE(NO)  | 0.43023255813953487 | 0.36363636363636365 | 0.35294117647058826 | 0.234375      |
+--------------+---------------------+---------------------+---------------------+---------------+
| FATIGUE(YES) | 0.5697674418604651  | 0.6363636363636364  | 0.6470588235294118  | 0.765625      |
+--------------+---------------------+---------------------+---------------------+---------------+
```

Figure 2: Conditional Probability Tables

that gave me this result:

LUNG_CANCER	phi(LUNG_CANCER)
LUNG_CANCER(NO)	0.1289
LUNG_CANCER(YES)	0.8711

I then computed by hand this result using the Variable Elimination algorithm taking into account the fact that, if I want to compute P(X | E), Y is an **irrelevant variable** if

$$Y \notin Ancestors(X \cup E)$$

In this case I have that every node under **LUNG_CANCER** is irrelevant as shown in Figure 3.

Knowing that

$$\mathbf{P}(X|\mathbf{e}) = \alpha\ \mathbf{P}(X, \mathbf{e}) = \sum_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y})$$

where each **y** is an **hidden variable**, and that I can compute a full joint query easily in a Bayesian Network by doing

$$P(x_1, ..., x_n) = \prod_{i=1}^{n} P(x_i|parents(X_i))$$

I ended up with the following formula:

$$\begin{aligned}
&\mathbf{P}(Lung\_cancer|peer\_pressure) \\
&\quad = \alpha \sum_{anx} \sum_{smoke} \sum_{age} \mathbf{P}(Lung\_cancer, peer\_pressure, anx, smoke, age) \\
&= \alpha \sum_{anx} \sum_{smoke} \sum_{age} P(peer\_pressure)\mathbf{P}(anx)\mathbf{P}(smoke|peer\_pressure, anx) \\
&\qquad\qquad \mathbf{P}(age)\mathbf{P}(Lung\_cancer|smoke, age) \\
&\quad = \alpha P(peer\_pressure) \sum_{anx} \mathbf{P}(anx) \sum_{smoke} \mathbf{P}(smoke|peer\_pressure, anx) \\
&\qquad\qquad\qquad \sum_{age} \mathbf{P}(age)\mathbf{P}(Lung\_cancer|smoke, age)
\end{aligned}$$

In order to apply the Variable Elimination Algorithm I assigned to each probability of the formula a function which depend only on the variables that haven't been summed up yet:

f_1(Anx) = **P**(anx)
f_2(S, Anx) = **P**(smoke|peer_pressure,anx)
f_3(Age) = **P**(age)
f_4(S, Age, L) = **P**(Lung_cancer|smoke,age)

I then computed the algorithm:

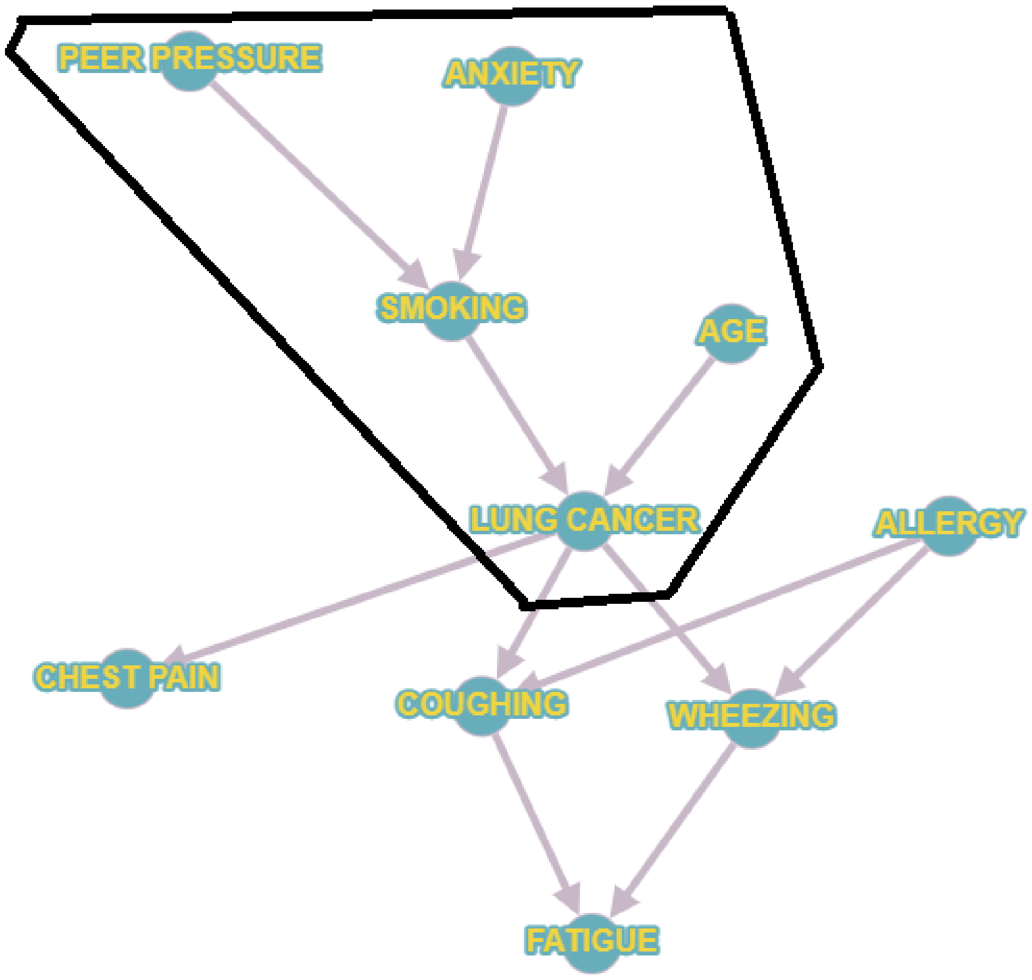


Figure 3: Irrelevant Variables for the query P(lung_cancer|peer_pressure: YES)

Age	S	T	f_5(S,Age,L)
0	T	T	0.0097 * 0.5 = 0.00485
0	T	F	0.0097 * 0.5 = 0.00485
0	F	T	0.0097 * 1.0 = 0.0097
0	F	F	0.0097 * 0.0 = 0.0
1	T	T	0.6246 * 0.8919 = 0.5571
1	T	F	0.6246 * 0.1081 = 0.06752
1	F	T	0.6246 * 0.8048 = 0.5027
1	F	F	0.6246 * 0.1951 = 0.1219
2	T	T	0.3657 * 0.9016 = 0.3297
2	T	F	0.3657 * 0.09836 = 0.03597
2	F	T	0.3657 * 0.9231 = 0.3376
2	F	F	0.3657 * 0.07692 = 0.02813

I computed f_5(S, Age, L) = f_3(Age) x f_4(S, Age, L) by looking at the CPTs. The result is shown in the following table:

I then did the sum-up step on Age resulting in f_6(S, L):

S	T	f_6(S,L)
T	T	0.00485 + 0.5571 + 0.3297 = 0.89165
T	F	0.00485 + 0.06752 + 0.03597 = 0.10834
F	T	0.0097 + 0.5027 + 0.3376 = 0.85
F	F	0.0 + 0.1219 + 0.02813 = 0.15

It continued with the computation of f_7(S,T,Anx) = f_2(S,Anx) x f_6(S,L):

S	T	A	f_7(S,T,Anx)
T	T	T	0.670212 * 0.89165 = 0.5976
T	F	T	0.670212 * 0.10834 = 0.07261
F	T	T	0.329787 * 0.85 = 0.28032
F	F	T	0.329787 * 0.15 = 0.04947
T	T	F	0.3442623 * 0.89165 = 0.30696
T	F	F	0.3442623 * 0.10834 = 0.0373
F	T	F	0.6557377 * 0.85 = 0.55738
F	F	F	0.6557377 * 0.15 = 0.09836

and the sum-up on S:

T	Anx	f_8(T,Anx)
T	T	0.5976 + 0.28032 = 0.8779
T	F	0.30696 + 0.55738 = 0.86434
F	T	0.07261+0.04947 = 0.12208
F	F	0.0373 + 0.09836 = 0.13566

I finally calculated the last product f_9(T,Aux) = f_1(Anx) x f_8(T,Anx):

T	Anx	f_9(T,Anx)
T	T	0.8779 * 0.498382 = 0.43753
T	F	0.86434 * 0.501618 = 0.43357
F	T	0.12208 * 0.498382 = 0.06084
F	F	0.13566 * 0.501618 = 0.06805

and the last summation:

T	f_10(T)
T	0.43753 + 0.43357 = 0.8711
F	0.06084 + 0.06805 = 0.12889

I finally computed

$$P(peer\_pressure) * f_{10}(T) =< 0.43696, 0.06465 >$$

and normalized the result:

$$\mathbf{P}(Lung\_cancer|peer\_pressure) =< \frac{0.43696}{0.50161}, \frac{0.06465}{0.50161} > \\ =< 0.8711, 0.1289 >$$

As we can see the result computed by hand **is the same as the one computed by pgmpy**.

4.2 Showing the Markov Blanket

We know from the theory that a node is **conditionally independent** from the others given his parents, his children and the parents of his children (Markov Blanket).

This is due to the fact that the query X is d-separated to everything else given the evidences E because they don't allow the creation of active paths bewteen X and the other nodes. This d-separation results in these nodes not being taken into account when the probability is calculated.

In order to show this conditional independence, I chose the attribute **'COUGHING'** and I calculated his markov blanket: **'LUNG_CANCER'** and **'ALLERGY'** (his parents), **'FATIGUE'** (his child), **'WHEEZING'** (the father of his child).

I then queried

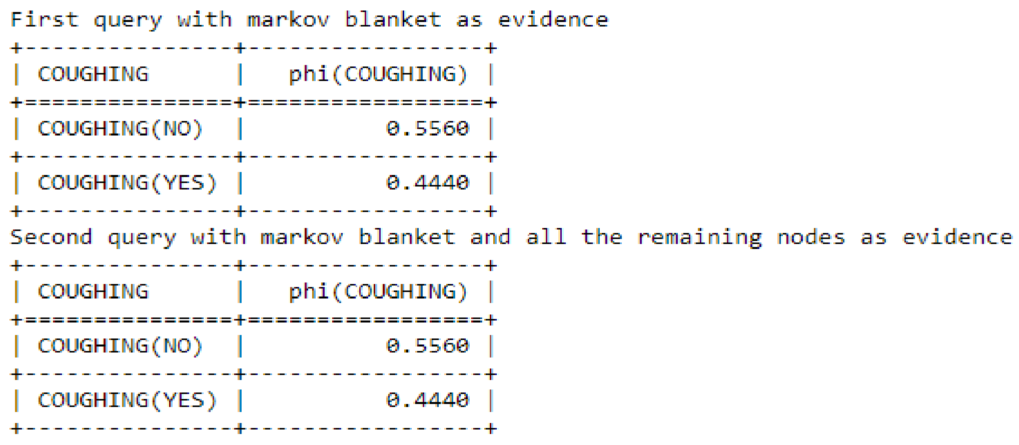
```
q2=lungcancer_infer.query(variables=['COUGHING'],
 evidence={'LUNG_CANCER':'YES','ALLERGY':'NO','FATIGUE':'NO',
 'WHEEZING':'YES'})
```

and

```
q3=lungcancer_infer.query(variables=['COUGHING'],
 evidence={'LUNG_CANCER':'YES','ALLERGY':'NO','FATIGUE':'NO',
 'WHEEZING':'YES','ANXIETY':'NO','AGE':2,'CHEST␣PAIN':'YES',
 'SMOKING':'YES','PEER_PRESSURE':'NO'})
```

and then I checked whether the result of the two queries was actually the same (Figure 4).

As expected the conditional independence coming from the markov blanket guarantees that new nodes in the evidence don't alter the result.



```
First query with markov blanket as evidence
+---------------+-----------------+
| COUGHING      |   phi(COUGHING) |
+===============+=================+
| COUGHING(NO)  |          0.5560 |
+---------------+-----------------+
| COUGHING(YES) |          0.4440 |
+---------------+-----------------+
Second query with markov blanket and all the remaining nodes as evidence
+---------------+-----------------+
| COUGHING      |   phi(COUGHING) |
+===============+=================+
| COUGHING(NO)  |          0.5560 |
+---------------+-----------------+
| COUGHING(YES) |          0.4440 |
+---------------+-----------------+
```

Figure 4: The tables resulting from queries q2 and q3. They are the same because of conditional independece.