

Implementation

```
!pip install pgmpy
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: pgmpy in /usr/local/lib/python3.7/dist-packages (0.1.20)
Requirement already satisfied: opt-einsum in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: joblib in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: pyparsing in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: networkx in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: statsmodels in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: scipy in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: torch in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: pandas in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: tqdm in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: patsy>=0.5 in /usr/local/lib/python3.7/dist-packages (from pgmpy)
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.7/dist-packages (from pgmpy)
```

```
from pgmpy.models import BayesianNetwork
from pgmpy.factors.discrete import TabularCPD
import networkx as nx
import pylab as plt
```

The BayesianModel can be initialized by passing a list of edges in the model structure. In this case, there are 4 edges in the model: Guest--> Host, Price-->host

```
# Defining Bayesian Structure
model = BayesianNetwork([('Guest', 'Host'), ('Price', 'Host')])
```

$P(C):$

C	0	1	2
	0.33	0.33	0.33

$P(P):$

P	0	1	2
	0.33	0.33	0.33

$P(H \mid P, C):$

C	0			1			2		
P	0	1	2	0	1	2	0	1	2
H=0	0	0	0	0	0.5	1	0	1	0.5
H=1	0.5	0	1	0	0	0	1	0	0.5
H=2	0.5	1	0	1	0.5	0	0	0	0

Defining the CPDs:

```
cpd_guest = TabularCPD('Guest', 3, [[0.33], [0.33], [0.33]])
cpd_price = TabularCPD('Price', 3, [[0.33], [0.33], [0.33]])
cpd_host = TabularCPD('Host', 3, [[0, 0, 0, 0, 0.5, 1, 0, 1, 0.5],
                                     [0.5, 0, 1, 0, 0, 0, 1, 0, 0.5],
                                     [0.5, 1, 0, 1, 0.5, 0, 0, 0, 0]],
                               evidence=['Guest', 'Price'], evidence_card=[3, 3])
```

Associating the CPDs with the network structure.

```
model.add_cpds(cpd_guest, cpd_price, cpd_host)
```

```
model.check_model()
```

True

Infering the posterior probability

```
from pgmpy.inference import VariableElimination
```

```
infer = VariableElimination(model)
posterior_p = infer.query(['Host'], evidence={'Guest': 2, 'Price': 2})
print(posterior_p)
```

```
+-----+-----+
| Host   | phi(Host) |
+=====+=====+
| Host(0) | 0.5000    |
+-----+-----+
| Host(1) | 0.5000    |
+-----+-----+
| Host(2) | 0.0000    |
+-----+-----+
```

```
# Infering the posterior probability
from pgmpy.inference import VariableElimination
```

```
infer = VariableElimination(model)
posterior_p = infer.query(['Host'], evidence={'Guest': 1, 'Price': 2})
print(posterior_p)
```

```
+-----+-----+
| Host   | phi(Host) |
+=====+=====+
| Host(0) | 1.0000    |
+-----+-----+
| Host(1) | 0.0000    |
+-----+-----+
| Host(2) | 0.0000    |
+-----+-----+
```

```
nx.draw(model, with_labels=True)
#plt.savefig('model.png')
#plt.close()
```

