

7. : Write a program to construct a Bayesian network considering medical data. Use this
- model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set.

```
!pip install bayespy
!pip install pgmpy
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: bayespy in /usr/local/lib/python3.9/dist-packages (0.5.25)
Requirement already satisfied: numpy>=1.10.0 in /usr/local/lib/python3.9/dist-packages (from bayespy) (1.22.4)
Requirement already satisfied: h5py in /usr/local/lib/python3.9/dist-packages (from bayespy) (3.1.0)
Requirement already satisfied: scipy>=0.13.0 in /usr/local/lib/python3.9/dist-packages (from bayespy) (1.10.1)
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting pgmpy
  Downloading pgmpy-0.1.21-py3-none-any.whl (1.9 MB)
    1.9/1.9 MB 20.6 MB/s eta 0:00:00
Requirement already satisfied: opt-einsum in /usr/local/lib/python3.9/dist-packages (from pgmpy) (3.3.0)
Requirement already satisfied: scipy in /usr/local/lib/python3.9/dist-packages (from pgmpy) (1.10.1)
Requirement already satisfied: statsmodels in /usr/local/lib/python3.9/dist-packages (from pgmpy) (0.13.5)
Requirement already satisfied: tqdm in /usr/local/lib/python3.9/dist-packages (from pgmpy) (4.65.0)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.9/dist-packages (from pgmpy) (1.2.2)
Requirement already satisfied: numpy in /usr/local/lib/python3.9/dist-packages (from pgmpy) (1.22.4)
Requirement already satisfied: joblib in /usr/local/lib/python3.9/dist-packages (from pgmpy) (1.1.1)
Requirement already satisfied: pandas in /usr/local/lib/python3.9/dist-packages (from pgmpy) (1.4.4)
Requirement already satisfied: torch in /usr/local/lib/python3.9/dist-packages (from pgmpy) (1.13.1+cu116)
Requirement already satisfied: networkx in /usr/local/lib/python3.9/dist-packages (from pgmpy) (3.0)
Requirement already satisfied: pyparsing in /usr/local/lib/python3.9/dist-packages (from pgmpy) (3.0.9)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.9/dist-packages (from pandas->pgmpy) (2022.7.1)
Requirement already satisfied: python-dateutil>=2.8.1 in /usr/local/lib/python3.9/dist-packages (from pandas->pgmpy) (2.8.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.9/dist-packages (from scikit-learn->pgmpy) (3.1.0)
Requirement already satisfied: packaging>=21.3 in /usr/local/lib/python3.9/dist-packages (from statsmodels->pgmpy) (23.0)
Requirement already satisfied: patsy>=0.5.2 in /usr/local/lib/python3.9/dist-packages (from statsmodels->pgmpy) (0.5.3)
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.9/dist-packages (from torch->pgmpy) (4.5.0)
Requirement already satisfied: six in /usr/local/lib/python3.9/dist-packages (from patsy->statsmodels->pgmpy) (1.15.0)
Installing collected packages: pgmpy
Successfully installed pgmpy-0.1.21
```

```
!pip install colorama
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: colorama in /usr/local/lib/python3.9/dist-packages (0.4.6)
```

```
import bayespy as bp
import numpy as np
import csv
from colorama import init
from colorama import Fore, Back, Style
init()
```

```
import pandas as pd
data=pd.read_csv("heartdisease.csv")
heart_disease=pd.DataFrame(data)
print(heart_disease)
```

	age	Gender	Family	diet	Lifestyle	cholesterol	heartdisease
0	0	0	1	1	3	0	1
1	0	1	1	1	3	0	1
2	1	0	0	0	2	1	1
3	4	0	1	1	3	2	0
4	3	1	1	0	0	2	0
5	2	0	1	1	1	0	1
6	4	0	1	0	2	0	1
7	0	0	1	1	3	0	1
8	3	1	1	0	0	2	0
9	1	1	0	0	0	2	1
10	4	1	0	1	2	0	1
11	4	0	1	1	3	2	0
12	2	1	0	0	0	0	0
13	2	0	1	1	1	0	1
14	3	1	1	0	0	1	0
15	0	0	1	0	0	2	1
16	1	1	0	1	2	1	1
17	3	1	1	1	0	1	0
18	4	0	1	1	3	2	0

## Importing Heart Disease Data Set and Customizing

```

from pgmpy.models import BayesianModel
model=BayesianModel([
('age','Lifestyle'),
('Gender','Lifestyle'),
('Family','heartdisease'),
('diet','cholesterol'),
('Lifestyle','diet'),
('cholesterol','heartdisease'),
('diet','cholesterol')
])

from pgmpy.estimators import MaximumLikelihoodEstimator
model.fit(heart_disease, estimator=MaximumLikelihoodEstimator)

from pgmpy.inference import VariableElimination
HeartDisease_infer = VariableElimination(model)

```

```

/usr/local/lib/python3.9/dist-packages/pgmpy/models/BayesianModel.py:8: FutureWarning: BayesianModel has been renamed to BayesianNe
warnings.warn(

```

```

print('For age Enter { SuperSeniorCitizen:0, SeniorCitizen:1, MiddleAged:2, Youth:3, Teen:4 }')
print('For Gender Enter { Male:0, Female:1 }')
print('For Family History Enter { yes:1, No:0 }')
print('For diet Enter { High:0, Medium:1 }')
print('For lifeStyle Enter { Athlete:0, Active:1, Moderate:2, Sedentary:3 }')
print('For cholesterol Enter { High:0, BorderLine:1, Normal:2 }')

```

```

q = HeartDisease_infer.query(variables=['heartdisease'], evidence={
    'age':int(input('Enter age :')),
    'Gender':int(input('Enter Gender :')),
    'Family':int(input('Enter Family history :')),
    'diet':int(input('Enter diet :')),
    'Lifestyle':int(input('Enter Lifestyle :')),
    'cholesterol':int(input('Enter cholesterol :'))
})

```

```
print(q['heartdisase'])
```

---

39 1 of 1 problem

---

*TypeError: 'DiscreteFactor' object is not subscriptable*

---

```

For age Enter { SuperSeniorCitizen:0, SeniorCitizen:1, MiddleAged:2, Youth:3, Teen:4 }
For Gender Enter { Male:0, Female:1 }
For Family History Enter { yes:1, No:0 }
For diet Enter { High:0, Medium:1 }
For lifeStyle Enter { Athlete:0, Active:1, Moderate:2, Sedentary:3 }
For cholesterol Enter { High:0, BorderLine:1, Normal:2 }
Enter age : 

```

## 7.2.2 Modeling Heart Disease Data

```

from pgmpy.models import BayesianModel
from pgmpy.estimators import MaximumLikelihoodEstimator, BayesianEstimator

model = BayesianModel([('age', 'trestbps'), ('age', 'fbs'), ('sex', 'trestbps'), ('sex', 'trestbps'),
    ('exang', 'trestbps'), ('trestbps', 'heartdisease'), ('fbs', 'heartdisease'),
    ('heartdisease', 'restecg'), ('heartdisease', 'thalach'), ('heartdisease', 'chol')])

```

```
# Learning CPDs using Maximum Likelihood Estimators
model.fit(heartDisease, estimator=MaximumLikelihoodEstimator)
#for cpd in model.get_cpds():
#    print("CPD of {variable}:".format(variable=cpd.variable))
#    print(cpd)

print(model.get_cpds('age'))
```

age(28)	0.00383142
age(29)	0.00383142
age(30)	0.00383142
age(31)	0.00766284
age(32)	0.0153257
age(33)	0.00766284
age(34)	0.0153257
age(35)	0.0191571
age(36)	0.0191571
age(37)	0.0306513
age(38)	0.0191571
age(39)	0.0344828
age(40)	0.0191571
age(41)	0.0383142
age(42)	0.0268199
age(43)	0.0421456
age(44)	0.0268199
age(45)	0.0229885
age(46)	0.045977
age(47)	0.0344828
age(48)	0.0613027
age(49)	0.0421456
age(50)	0.045977
age(51)	0.0344828
age(52)	0.0574713
age(53)	0.0383142
age(54)	0.0842912
age(55)	0.0536398
age(56)	0.0306513

```
print(model.get_cpds('chol'))
```

heartdisease	heartdisease(0)	heartdisease(1)
chol(100)	0.006134969325153374	0.0
chol(117)	0.0	0.01020408163265306
chol(129)	0.006134969325153374	0.0
chol(132)	0.006134969325153374	0.0
chol(147)	0.012269938650306749	0.0
chol(156)	0.0	0.01020408163265306
chol(160)	0.012269938650306749	0.01020408163265306
chol(161)	0.006134969325153374	0.0

chol(163)	0.006134969325153374	0.0
chol(164)	0.0	0.01020408163265306
chol(166)	0.006134969325153374	0.0
chol(167)	0.006134969325153374	0.0
chol(168)	0.006134969325153374	0.0
chol(171)	0.006134969325153374	0.0
chol(172)	0.0	0.01020408163265306
chol(173)	0.006134969325153374	0.0
chol(175)	0.0	0.01020408163265306
chol(179)	0.012269938650306749	0.0
chol(180)	0.006134969325153374	0.01020408163265306
chol(182)	0.012269938650306749	0.01020408163265306
chol(184)	0.018404907975460124	0.0
chol(186)	0.006134969325153374	0.01020408163265306
chol(187)	0.006134969325153374	0.0
chol(188)	0.012269938650306749	0.0
chol(190)	0.006134969325153374	0.0
chol(193)	0.006134969325153374	0.02040816326530612
chol(194)	0.012269938650306749	0.0
chol(195)	0.018404907975460124	0.0

```
print(model.get_cpds('sex'))
```

sex(0)	0.264368
sex(1)	0.735632

```
model.get_independencies()
```

```
(age _|_ exang, sex)
(age _|_ sex | exang)
(age _|_ exang, sex | fbs)
(age _|_ thalach, chol, restecg | heartdisease)
(age _|_ exang | sex)
(age _|_ thalach, restecg | chol, heartdisease)
(age _|_ thalach, chol, heartdisease, restecg | trestbps, fbs)
(age _|_ thalach, chol, restecg | trestbps, heartdisease)
(age _|_ chol, restecg | thalach, heartdisease)
(age _|_ sex | exang, fbs)
(age _|_ thalach, chol, restecg | exang, heartdisease)
(age _|_ thalach, chol, restecg | fbs, heartdisease)
(age _|_ exang | fbs, sex)
(age _|_ thalach, chol, restecg | heartdisease, sex)
(age _|_ thalach, chol | heartdisease, restecg)
(age _|_ thalach, heartdisease, restecg | trestbps, chol, fbs)
(age _|_ thalach, restecg | trestbps, chol, heartdisease)
(age _|_ restecg | thalach, chol, heartdisease)
(age _|_ thalach, restecg | exang, chol, heartdisease)
(age _|_ thalach, restecg | fbs, chol, heartdisease)
(age _|_ thalach, restecg | chol, heartdisease, sex)
(age _|_ thalach | chol, heartdisease, restecg)
(age _|_ chol, heartdisease, restecg | trestbps, thalach, fbs)
(age _|_ chol, restecg | trestbps, thalach, heartdisease)
(age _|_ thalach, chol, heartdisease, restecg | trestbps, fbs, exang)
(age _|_ thalach, chol, restecg | trestbps, heartdisease, exang)
(age _|_ thalach, chol, restecg | trestbps, fbs, heartdisease)
(age _|_ thalach, chol, heartdisease, restecg | trestbps, fbs, sex)
(age _|_ thalach, chol, heartdisease | trestbps, fbs, restecg)
(age _|_ thalach, chol, restecg | trestbps, heartdisease, sex)
(age _|_ thalach, chol | trestbps, heartdisease, restecg)
(age _|_ chol, restecg | thalach, heartdisease, exang)
(age _|_ chol, restecg | thalach, fbs, heartdisease)
(age _|_ chol, restecg | thalach, heartdisease, sex)
(age _|_ chol | thalach, heartdisease, restecg)
(age _|_ thalach, chol, restecg | exang, fbs, heartdisease)
(age _|_ thalach, chol, restecg | exang, heartdisease, sex)
(age _|_ thalach, chol | exang, heartdisease, restecg)
(age _|_ thalach, chol, restecg | fbs, heartdisease, sex)
```

```
(age | thalach, chol | fbs, heartdisease, restecg)
(age | thalach, chol | heartdisease, restecg, sex)
(age | heartdisease, restecg | trestbps, chol, thalach, fbs)
(age | restecg | trestbps, chol, heartdisease, thalach)
(age | thalach, heartdisease, restecg | trestbps, chol, exang, fbs)
(age | thalach, restecg | trestbps, chol, heartdisease, exang)
(age | thalach, restecg | trestbps, chol, heartdisease, fbs)
(age | thalach, heartdisease, restecg | trestbps, chol, sex, fbs)
(age | thalach, heartdisease | trestbps, chol, restecg, fbs)
(age | thalach, restecg | trestbps, chol, heartdisease, sex)
(age | thalach | trestbps, chol, heartdisease, restecg)
(age | restecg | thalach, chol, heartdisease, exang)
(age | restecg | thalach, chol, heartdisease, fbs)
(age | restecg | thalach, chol, heartdisease, sex)
(age | thalach, restecg | exang, chol, heartdisease, fbs)
(age | thalach, restecg | exang, chol, heartdisease, sex)
(age | thalach | exang, chol, heartdisease, restecg)
(age | thalach, restecg | fbs, chol, heartdisease, sex)
(age | thalach | fbs, chol, heartdisease, restecg)
```

7.2.3.Inferencing with Bayesian Network

```
# Doing exact inference using Variable Elimination
from pgmpy.inference import VariableElimination
HeartDisease_infer = VariableElimination(model)

# Computing the probability of bronc given smoke.
q = HeartDisease_infer.query(variables=['heartdisease'], evidence={'age': 28})
print(q['heartdisease'])
```

heartdisease	phi(heartdisease)
heartdisease_0	0.6333
heartdisease_1	0.3667

```
q = HeartDisease_infer.query(variables=['heartdisease'], evidence={'chol': 100})
print(q['heartdisease'])
```

heartdisease	phi(heartdisease)
heartdisease_0	1.0000
heartdisease_1	0.0000