AI Assignment - 8

Bayesian Network

Sabarivasan Velayutham

205001085

CSE-B

Code:

```
import math


# PART B
# Finding conditional probability expression for every random variable



def expression(rv, exp):
    if len(network[rv]) == 0:
        return 'P(' + rv + ')'
    else:
        cond = ''
        cond += network[rv][0]
        for i in range(1, len(network[rv])):
            cond += ',' + network[rv][i]
        exp += 'P(' + rv + '|' + cond + ')'
        for v in network[rv]:
            exp += expression(v, exp)
    return exp
```

```
# PART B
# Finding overall conditional probability expression



def total_exp(network, cond_pr, exp):
    for k in network:
        if len(network[k]) == 0:
            cond_pr[k] = 'P(' + k + ')'
        else:
            cond = ''
            cond += network[k][0]
            for i in range(1, len(network[k])):
                cond += ',' + network[k][i]
            cond_pr[k] = 'P(' + k + '|' + cond + ')'
        exp += cond_pr[k]
    return exp


# PART C
# Finding number of independent parameters required with conditional
independence



def ind_param(network, cond_pr, total):
    for k in network:
        param = int(math.pow(2, len(network[k])))
        print(cond_pr[k] + '\t\t\t' + str(param))
        total += param
    return total
```

```
# PART D
# Finding number of independent parameters required with no conditional
independence



def no_con_ind_param(network):
    no = len(network)
    print('\nNumber of independent parameters if there is no
conditional independence = ' +
          '2^' + str(no) + ' - 1' + ' = ' + str(math.pow(2, no) - 1))


# PART E
# Markov Blanket



def markov_blanket(network, rv, markov):
    markov.append(rv)
    for v in network[rv]:
        if v not in markov:
            markov.append(v)


    for k in network:
        if rv in network[k]:
            if k not in markov:
                markov.append(k)
            for v in network[k]:
                if v not in markov:
                    markov.append(v)
```

```
# PART A

network = {'D1': [], 'D2': [], 'D3': [], 'S1': ['D1'],

           'S2': ['D1', 'D2'], 'S3': ['D1', 'D3'], 'S4': ['D3']}


print('CONDITIONAL PROBABILITIES')

for k in network:

    print(k + ': ' + expression(k, ''))


cond_pr = {}

print('Overall expression: ' + total_exp(network, cond_pr, ''))


print('\nCPT\t\t\t\tNumber of Independent Parameters')

total = ind_param(network, cond_pr, 0)

print('Total\t\t\t' + str(total))


no_con_ind_param(network)


print('\nMARKOV BLANKET')

for k in network:

    markov = []

    markov_blanket(network, k, markov)

    print(k + ': ' + str(markov))
```

Output:

```
PS C:\Users\sabar\OneDrive\Desktop\LAB\Artificial Intelligence> python -u "c:\Users\sabar\On
eDrive\Desktop\LAB\Artificial Intelligence\EX-8 Bayesian Network\bayesian.py"
CONDITIONAL PROBABILITIES
D1: P(D1)
D2: P(D2)
D3: P(D3)
S1: P(S1|D1)P(D1)
S2: P(S2|D1,D2)P(D1)P(D2)
S3: P(S3|D1,D3)P(D1)P(D3)
S4: P(S4|D3)P(D3)
Overall expression: P(D1)P(D2)P(D3)P(S1|D1)P(S2|D1,D2)P(S3|D1,D3)P(S4|D3)

CPT                             Number of Independent Parameters
P(D1)                   1
P(D2)                   1
P(D3)                   1
P(S1|D1)                        2
P(S2|D1,D2)                     4
P(S3|D1,D3)                     4
P(S4|D3)                        2
Total                   15

Number of independent parameters if there is no conditional independence = 2^7 - 1 = 127.0

MARKOV BLANKET
D1: ['D1', 'S1', 'S2', 'D2', 'S3', 'D3']
D2: ['D2', 'S2', 'D1']
D3: ['D3', 'S3', 'D1', 'S4']
S1: ['S1', 'D1']
S2: ['S2', 'D1', 'D2']
S3: ['S3', 'D1', 'D3']
S4: ['S4', 'D3']
PS C:\Users\sabar\OneDrive\Desktop\LAB\Artificial Intelligence>
```