Bhuvi Ghosh
60009210191

In [ ]:

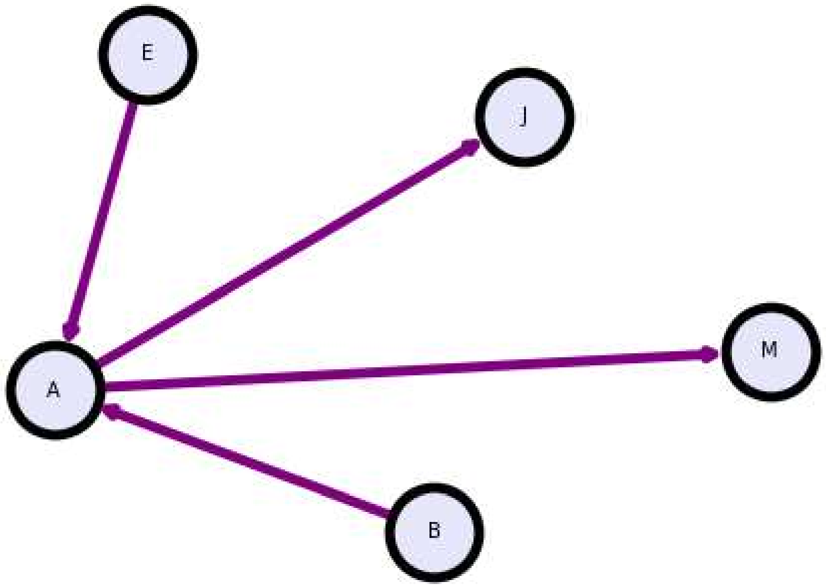
```
from pgmpy.models import BayesianNetwork
from pgmpy.factors.discrete import TabularCPD
# Defining the network structure
model = BayesianNetwork([("B", "A"), ("E", "A"), ("A", "J"), ("A", "M")])
# Defining the CPDs:
cpd_b = TabularCPD("B", 2, [[0.999], [0.001]])
cpd_e = TabularCPD("E", 2, [[0.998], [0.002]])
cpd_a = TabularCPD("A",2, [[0.999,0.71,0.06,0.05], [0.001,0.29,0.94,0.95]], evide
cpd_j = TabularCPD("J",2, [[0.95,0.10],[0.05,0.90]], evidence=["A"], evidence_car
cpd_m = TabularCPD("M",2, [[0.99,0.30],[0.01,0.70]], evidence=["A"], evidence_car
# Associating the CPDs with the network structure.
model.add_cpds(cpd_b, cpd_e, cpd_a, cpd_j, cpd_m)
# Some other methods
model.get_cpds()
```

Out[26]:

```
[<TabularCPD representing P(B:2) at 0x7fa5f8cd2550>,
 <TabularCPD representing P(E:2) at 0x7fa5f87cfb10>,
 <TabularCPD representing P(A:2 | B:2, E:2) at 0x7fa5f87cfe50>,
 <TabularCPD representing P(J:2 | A:2) at 0x7fa5f87cf590>,
 <TabularCPD representing P(M:2 | A:2) at 0x7fa5f87cf7d0>]
```

In [ ]:

```
import networkx as nx
import matplotlib.pyplot as plt
options = {
"font_size": 10,
"node_size": 2000,
"node_color": "lavender",
"edgecolors": "black",
"edge_color": "purple",
"linewidths": 5,
"width": 5,}
nx.draw(model, **options, with_labels=True)
plt.show()
```



In [ ]:

```
print(cpd_b)
print(cpd_e)
print(cpd_a)
print(cpd_j)
print(cpd_m)
```

```
+------+-------+
| B(0) | 0.999 |
+------+-------+
| B(1) | 0.001 |
+------+-------+
+------+-------+
| E(0) | 0.998 |
+------+-------+
| E(1) | 0.002 |
+------+-------+
+------+-------+------+------+------+
| B    | B(0)  | B(0) | B(1) | B(1) |
+------+-------+------+------+------+
| E    | E(0)  | E(1) | E(0) | E(1) |
+------+-------+------+------+------+
| A(0) | 0.999 | 0.71 | 0.06 | 0.05 |
+------+-------+------+------+------+
| A(1) | 0.001 | 0.29 | 0.94 | 0.95 |
+------+-------+------+------+------+
+------+------+------+
| A    | A(0) | A(1) |
+------+------+------+
| J(0) | 0.95 | 0.1  |
+------+------+------+
| J(1) | 0.05 | 0.9  |
+------+------+------+
+------+------+------+
| A    | A(0) | A(1) |
+------+------+------+
| M(0) | 0.99 | 0.3  |
+------+------+------+
| M(1) | 0.01 | 0.7  |
+------+------+------+
```

In [ ]:

```
# Initializing the VariableElimination class
from pgmpy.inference import VariableElimination
infer = VariableElimination(model)
```

In [ ]:

```
q = infer.query(["M"],{"B": 1})
print(q)
```

```
+------+----------+
| M    |   phi(M) |
+======+==========+
| M(0) |   0.3414 |
+------+----------+
| M(1) |   0.6586 |
+------+----------+
```

In [ ]:

```
q = infer.query(["J"],{"M": 1})
print(q)
```

```
+------+----------+
| J    |   phi(J) |
+======+==========+
| J(0) |   0.8224 |
+------+----------+
| J(1) |   0.1776 |
+------+----------+
```