

Source code Overview and Structure

- Programming Language: Python
- Version: 3.6
- *The code file is. ipynb (Jupyter notebook), no additional package installation is required.
- The code is as per pseudocode.
- It accepts both binary and non-binary values for each node.
- Compatible for loop less graphs.
- It consists of 4 major functions.
- Initialize network, Instantiate network, Lambda Message, PI Message

Instructions for Code Flow

Please follow below format to check the functionality of the code

The code expects below (9) inputs and is read from a text file :

- 1.Number of nodes present in the graph "N".
- 2.List all "N" nodes.
- 3.Number of edges "M" present between "N" nodes. (Directed Edge)
- 4.List all "M" edges in order i.e, Parent--->Child (First enter parent, give a space then enter its corresponding child. If more than one child re enter the edge with parent again)
- 5.Enter all possible values that a node can take. Format: Node(space)value1(space)value2(space)....so on. Ex: [A A0 A1] (Here node A can take value 0 and value 1, note that value is mentioned along with node to reduce code complexity retaining the meaning)
- 6.Number of marginal Probabilities given.
- 7.Enter all marginal Probabilities. Ex: A1 0.7 (This means $P(A1) = 0.7$, give a space after node value)
- 8.Number of conditional Probabilities.
- 9.Enter all Conditional Probabilities. Ex: C1 B1A1 0.1 (This means $P(C1 | B1 A1) = 0.1$, give a space after Node value. Also, please note that code interprets C1 B1A1 and C1 A1B1 as two different case though logically they are same, hence kindly input same value by interchanging the evidence set)

1.Graph

2.All probabilities asked in 1b of final project.

*To instantiate any other case, we need to pass as arguments to instantiate network function.