Introduction to Artificial Intelligence – summer 2022 – task #7

Version 1.0

Teacher: mgr inż. Maciej Szymkowski, maciej.szymkowski@pw.edu.pl

Please be aware that this exercise is done in pairs (you should inform us about the groups via Teams #General channel – under the post of prof. Paweł Wawrzyński).

Assignment description

This time we are dealing with the idea of Bayesian networks. We would like to observe how they work and what kind of information they can provide us. To appropriately deal with this task you need to implement simple software that can perform basic inference in Bayesian networks with MCM algorithm and Gibbs sampling.

The program should be implemented in the form of a console application (no Graphical User Interface is needed). The functionalities that are required are as follows:

- Your program should be able to read a Bayesian network defined in the given JSON file (please look at attached flu.json file).
- The software should also be able to print out the Bayesian network structure – all nodes need to be given as well as Markov blanket for selected variables should be visible.
- It also needs to accept evidence – which sets the observed variables of specific nodes.
- The software should also be able to answer on simple queries – for instance, it should return the probability distribution of the selected query variables.
- Moreover, it should allow user to set the number of steps performed by MCMC algorithm.

flu.json – file structure – definitions

- ***nodes*** – define all the variable names within the network; each name is an alphanumerical string with possible underscores (so we need to keep in mind that the possible characters are A…Za…z0….9_)
- ***relations*** – define the edges within the network along with the probability tables
- Please also look at the order of the keys in the sections ParentValues, ChildValues – for example "T, F": 0.10 in the HighFever means $\mathrm{P(HighFever{=}T|Flu{=}F)} = 0.10$

Technical requirements

In the case of this task you need to implement the logic that will not only perform all required functionalities but also can verify whether the provided Bayesian network is correct. It means that the network cannot have cycles in the graph, all nodes must have defined probabilities as well as the probability tables are correct (sum of all probabilities in each case must be equal 1).

Please, be aware that all logic needs to be implemented directly by you. **It is not allowed to use any ready-to-use machine learning or optimization libraries / packages (e.g., PyTorch, TensorFlow or even PyMC3).** It means that the task is to implement whole logic by yourself (of course, you can use basic linear algebra and math processing packages / libraries – e.g., NumPy).

The preferred language to solve this task is Python. However, it is still possible to prepare the solution with Java/Scala/C/C++/R/C# languages.

How-to submit the task?

- **Deadline:** You should submit the task (source code of your solution) to maciej.szymkowski@pw.edu.pl. The deadline for Wednesday's group is 15.06.2022, 11:59:59 CET.

- In the title of your e-mail please include “[EARIN] Exercise #7”. Moreover, do not forget to include names, surnames, and e-mails of all members of your group (please add them in the e-mail content).
- The results will be discussed during next meeting. The details related to it will be given within our Microsoft Teams channel.
- You can get 0-7 points for this task.
- If you have any questions, please do not hesitate to contact me via e-mail (maciej.szymkowski@pw.edu.pl) or directly by Microsoft Teams platform.