Bayesian nets
Meeting notes 1/27/2023

Everybody was there, either physically or virtually.

I. Unfortunately, Patrick monopolized the first half of the meeting. The points made:

A. In contrast to an earlier version, there **is** an algorithm for generating a random non-looped network of a given set of nodes:

Form a random partition of the set of nodes.
Assign a 'level' at random to each partition.
Starting at the highest (lowest-numbered) level, create a directed link from each node to one or more nodes on a lower level (or connect it to nothing). Go to next node and repeat.

This will always form a non-looped network, since no directed link will go 'up' to complete a loop. By the discussion in Mark Newman's pages, every non-looped network of the given number of nodes will be captured by this.

[As an afterthought, Patrick isn't sure that it might not be redundantly captured: i.e., the same network generated by different partitions, level-assignments, and-or directed-link wirings. Given redundancy, the chance of different networks being generated at random may not be the same. Hmm. Well, let's put that aside.]

B. Zhongming has done a great calculation of the sample space of directed networks with n nodes, which seems both theoretically right and is giving the right numbers through 6 nodes. His paper on that is attached.

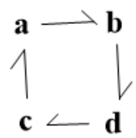
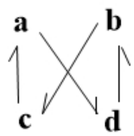
Zhonghing seemed to have some doubts about the general formula giving non-integer values for higher n. I don't understand that yet.

C. Zhongming has said that his calculation includes all directed networks, but doesn't eliminate the loops. Patrick suggested a way to eliminate the loops, though he now notes a problem with it:

There will be no loops for networks in n nodes with just one directed link, or two.

For three directed links:
How many 'triangles' are there? $\binom{n}{3}$. For each one, a 'circle' can go one of two ways. Those are all of the 3-node loops. So you'll subtract 2 * $\binom{n}{3}$ from Zhongming's calculation.

For four directed links:
How many 'squares' are there? $\binom{n}{4}$. For each of those, how many ways to loop? In the meeting, Patrick acted as if there were only 2. But that doesn't seem right. Consider:

So not as simple as Patrick thought. For each tuple, will have to figure out how many ways of looping with precisely that many nodes (not including a loop already with a sub-graph).

If we could make this work, the hope is that we could convert Zhongming's work from just a 'how many' calculation to one that spit out a coding of some kind for all the non-looped directed networks of a given n nodes. With that we could form a look-up table that would make it easy (a) to 'pick a random non-looped network of n nodes' and (b) ask whether a newly formed network was looped or not. Zhongming is going to take this as a further goal as well.

D. Neither the 'generating a random non-looped network' idea in (A) above, nor the 'subtracting looped networks from Zhongming's formula' idea (short of the look-up table) address the generation of a new network by mutation or hybridization and having to check whether the result is looped or not. Amber says that we have basically just 'asked pyAgrum' to this point.

There **is** an algorithm for checking whether a network has a loop, again taken from Mark Newman's pages. For all we know, this may be what Pyagrum uses. As Amber noted, implementing it might be just as expensive in time. The look-up table would still be preferable.

Here is the Mark Newman algorithm to tell whether any given network has a loop:

(1) Are there any nodes that have nothing coming from them? If not, it's looped.
(2) If it does have one or more nodes with nothing coming from them, eliminate one together with links to it.
Now consider the sub-network that remains, and go back to (1).

If at any point you reach a sub-network with links and in which there are no nodes with nothing coming from them, the original network had a loop. If you keep erasing nodes and links until nothing is left, that network had no loops.

II. Finally we got to stuff that wasn't just Patrick's mental download, complete with its mistakes.

Dennis has produced a new representation of evidence that avoids the problem of two networks, both of which score '1', but only one of which really matches the world. The problem was that our evidence-scoring indicated whether activation of node a produced activation of node b, but failed to distinguish activating b at different time-steps or by way of multiple routes.

Dennis' new representation of evidence by steps avoids that problem. Amber pointed out that scoring how well a theory matched the world was no longer as easy as it was before. With that new evidence representation it appears we need a new scoring method. Application to 'keep the best' strategy seemed particularly puzzling.

Dennis is going to think about how we should do scoring with the new evidence method.

In the course of that discussion, we also noted that we might want a scoring method that reflected the possibility that activation at one node came out great in term of a match, but activation at another node did not. We may not want merely an average across scores from different activation points as our 'score,' since it might be very informative where the problem occurred.

III. Patrick also suggested a new 'hill-climbing' heuristic for changing the structure of your theory in the attempt to match the world:

Start with a given network, complete with an assignment of levels.
To explore a variation:

a. Erase or add a link by the 'from a node to something on a lower level' idea in I.A. above.
b. Or: pick a node, erase all links, put it on another level and add random links to and from it in accord with the 'directed links to lower level' routine.

Patrick pointed out that, like many hill-climbing methods, this might well end up with 'local maxima': your theory isn't the best, but any small changes from where you are would make it worse, and thus you can't reach the higher peak by progressive positive steps. Sophia asked whether there are any algorithms to help with that. The only ones Patrick knows about are 'okay, try a radical change' leaps. A linkage between multiple agents each of which is exploring a different territory can also help—if shared evidence indicates that someone is hill-climbing a higher hill, you can switch over there.

IV. We started a discussion on making one or more of our heuristics plausibly match the psychological and/or normative question of 'well, if your theory isn't quite right, how would you (or should you) explore changes?

With regard to (III), Sophia also mentioned a recent book by Neil Levy, **Bad Beliefs: Why They Happen to Goiod People** that may be relevant in terms of belief-forming heuristics that trap one in conspiracy theories and the like.

V. For next week: Zhongming has plenty of challenges as outlined above.
Dennis is going to think about options for a new version of scoring appropriate to our new version of evidence flow.
Amber can cool it until that comes in, but she'll probably be exploring stuff anyway.
Patrick thinks we're moving toward paper-like results here, and will think more about what a paper like that would look like.

Same time next week, probably our original room.