IJCAI 2019 Tutorial

Medical decision analysis with probabilistic graphical models

Presenter: Francisco Javier Díez, PhD
Dept. Artificial Intelligence, UNED, Spain
fjdiez@dia.uned.es.

Hands-on exercises to be done with OpenMarkov during the tutorial. You must download the file exercises.zip and uncompress it where you can easily find the files it contains. We strongly recommend you not to look at the solutions before trying to solve the exercises yourself.

We also recommend you to read through the **hints** we offer for each exercise before you start reproducing them in OpenMarkov.

1. Probabilistic diagnosis

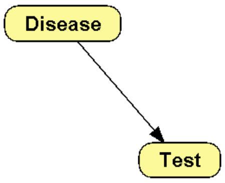
Exercise

The prevalence of a disease in a certain subpopulation is 14%. There is a test for this disease with a sensitivity of 70% and a specificity of 91%.

- What is the positive predictive value of the test?
- What is the negative predictive value?

Hints

- We will create a Bayesian network with two nodes, *Disease* and *Test*, and introduce evidence on the second in order to compute the posterior probabilities of the disease given the test result.
- Create a new network with button 🗋. Check that Network type is *Bayesian network*.
- Create a chance node by clicking on button ○ and then clicking where you wish to place it. Double-click on this node to open the Node properties dialog and change its name to *Disease*. In the Domain tab, check that the states are *present* and *absent*. Click OK.
- Create the node *Test* similarly. In the Domain tab, assign the states *positive* and *negative*, either by writing the names in the corresponding cells or by selecting them with the Standard domains button.
- Select the button ↗ and draw a link by dragging the pointer from node *Disease* to node *Test*, as shown in the next figure.



- If you wish to move a node, select it with button ↖ and drag it.
- Edit the probability table for *Disease*, either by alt-clicking on this node or by right-clicking on it and selecting the Edit probability option. In the upper cell, write *0.14*, the prevalence given in the statement of the problem, which is the probability of the disease being present.

- Edit the probability table for *Test*. In the upper right cell, write the sensitivity, 0.70, which is the probability of a positive result when the disease is present. In the lower left cell write the specificity, 0.91, which is the probability of a negative result when the disease is absent.
- Save the network as *BN-one-test.pgmx*, checking that the file type is *ProbModel 0.2 (*.pgmx)*.
- Compile the network with the ⚡ button.
- Introduce the finding *Test* = *positive* by double-clicking on the state *positive* of node *Test*. The positive predictive value is the probability of *Disease* = *present*. This will answer the first question in the statement of the problem.
- Introduce the finding *Test* = *negative*. The negative predictive value is the probability of *Disease* = *absent*. This will answer the second question.

2. Learning Bayesian networks

Exercise

- Learn different Bayesian networks from the dataset *asia10K.csv*, contained in file exercises.zip, using two algorithms: *hill-climbing* and *PC*.
- This dataset contains 10,000 fictitious patients. They were randomly generated by probabilistic sampling from the network *BN-asia.pgmx*, also included in exercises.zip. We will learn a Bayesian network as if we didn't know the true model that generated the data.

Hints

- We recommend that you first open the network *BN-asia.pgmx* (the "true" model) with OpenMarkov and then the dataset *asia10K.csv* with a text editor, to gain some insight about it. The first line of the dataset contains the names of the variables and each of the following lines contains the values for a case (a patient).

Learn a Bayesian network with the hill-climbing algorithm

- In OpenMarkov, go to Tools > Learning and select the dataset *asia10K.csv*. Check that, by default, the algorithm is *hill-climbing* and the metric is *K2*. The alpha parameter (Laplace-like correction) will be used when estimating the parameters of the network, not when learning the structure.
- Click Learn. By default, OpenMarkov will place the nodes in a circle.
- You can accept the edits proposed by OpenMarkov or choose your owns, either by selecting one from the list or by drawing/removing links from the graph. Please note that every time the network is modified, the list of edits is updated.
- Save the network learned if you wish.

Learn a Bayesian network with the PC algorithm

- Go again to Tools > Learning and replace Hill climbing with PC. Check that initially the significance level is 0.05.
- Click Learn. Accept one by one all the edits proposed by OpenMarkov. Observe that the algorithm has three phases: removing links, orienting pairs of links that converge on a node, and orienting the rest of the links.
- Apply again the PC algorithm for different values of the significance level—for example, 0, 0.001, 0.01, 0.1, 0.5, 0.7, 0.9, 0.99, and 1. You can set it by clicking the Options button after selecting the PC algorithm. Count the number of links in the resulting network for each significance level.

3. Unicriterion decision analysis

Exercise

- For a certain disease, whose prevalence is 2%, there are two tests, with the following properties:

Test	sensitivity	specificity	discomfort	cost
A	0.60	0.92	0.0003 QALY	$100
B	0.80	0.91	0.0001 QALY	$200

- There is also a therapy, costing $7,000. The effectiveness is:

Disease →	absent	present
therapy	38 QALY	30 QALY
no therapy	40 QALY	20 QALY

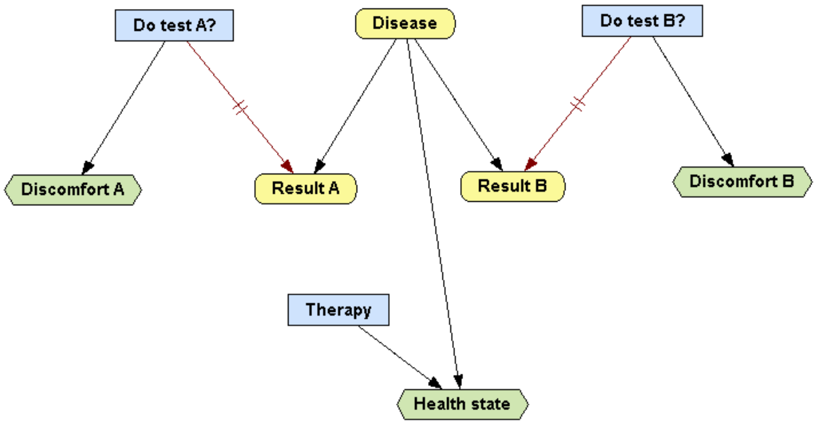
- What is the most effective strategy?

Hints

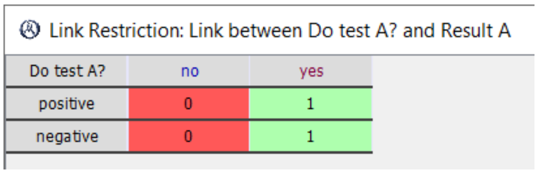
- Given that the goal is to maximize the effectiveness, we disregard costs at the moment. (We will consider them in the next exercise, which is a cost-effectiveness analysis for the same problem.)

Create the structure of the network (including only test A)

- Click the button 🗋 to create a new model. Network type must be *Decision analysis network*.
- Right-click anywhere on the empty panel to open the Network properties dialog. Go to tab Advanced, select Decision criteria, Standard criteria, and *Cost-effectiveness ($/QALY)*.
- Create the chance node *Disease*; its domain will have the default values, *present* and *absent*.
- Create the chance node *Result A*, with domain {*positive*, *negative*}. You may use the Standard domains button.
- Create the decision node *Do test A?*. Check that its domain is {*yes*, *no*}, the default for decision nodes.
- Create the utility node *Discomfort A*. In the Network properties dialog, the Decision criterion must be "*Effectiveness (QALY)*".
- Create the decision *Therapy*.
- Create the utility node *Health state*, whose Decision criterion is also *effectiveness*.
- Draw the causal links, as in the next figure (even though you have not yet added the 3 nodes for Test B). You can use the button ↖ to select and move the nodes and the button ↗ to draw links.



- Save the network so that you don't lose your work by accident. The name can be, for example, *DAN-two-tests.pgmx*.
- Right-click on the link *Do test A?* → *Result A* and select Add restrictions. Mark the two cells in the left column of the table to indicate that when the test is not done, the result is neither *positive* nor *negative*. The table will look as follows:



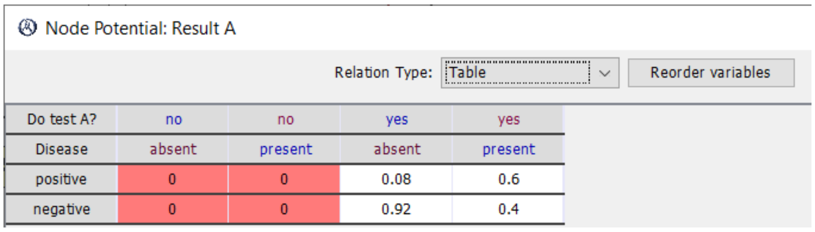
Link Restriction: Link between Do test A? and Result A

Do test A?	no	yes
positive	0	1
negative	0	1

- Right-click again on the link *Do test A?* → *Result A*, select Edit revelation conditions, and mark the checkbox for *yes* to indicate that when the test is done, the result (either positive or negative) is known.

Introduce the numerical parameters

- Open the probability table for *Disease*. Remember that you can do it by alt-clicking or right-clicking on the node. Introduce the prevalence, 0.02.
- Open the probability table for *Result A*. Introduce the sensitivity and specificity of the test, as shown below. You might need to reorder the parents with the Reorder variables button so that the row for *Do test A* appears above the row for *Disease*.

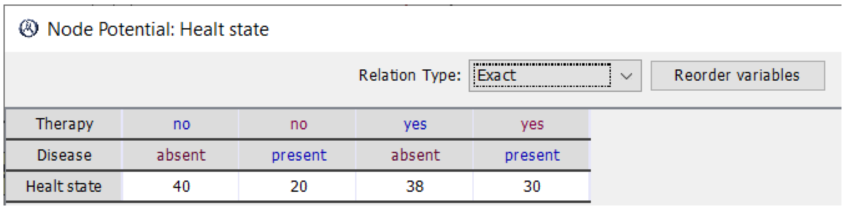


Node Potential: Result A

Relation Type: Table

Do test A?	no	no	yes	yes
Disease	absent	present	absent	present
positive	0	0	0.08	0.6
negative	0	0	0.92	0.4

- Open the utility table for *Health state*. Set Relation type to *Exact* and introduce the numeric values, as follows:



Node Potential: Healt state

Relation Type: Exact

Therapy	no	no	yes	yes
Disease	absent	present	absent	present
Healt state	40	20	38	30

- Open the utility table for *Discomfort A*. Set Relation type to *Exact* and introduce the numeric values: when the test is done, the discomfort is –0.0003 QALY (a negative effectiveness); when the test is not done, there is no discomfort.
- Save the network again.

Evaluate the model

- Click the button to expand the decision tree. Observe that the option Unicriterion is selected. It means that OpenMarkov will convert cost and effectiveness into a single criterion. In this example there is no cost, so the only criterion will be the effectiveness. Therefore, you must select *QALY* as the unit. Click OK.
- The optimal strategy can be obtained be inspecting the tree and selecting for each decision node the branch that maximizes the effectiveness, which is marked with a red rectangle.

- Another way to get the optimal policy (i.e., an alternative to expanding the decision tree and inspecting it) is to click the S button: OpenMarkov will show the optimal strategy as an intervention tree, which is much more compact than the decision tree.

Add test B and evaluate the model again

- Add now the three nodes (chance, decision, and utility) for test B, as you did for test B.
- Save the network as *DAN-two-tests-without-costs.pgmx*.
- If you do not have enough time, open the network *DAN-two-tests-without-costs.pgmx*, available in exercises.zip, in the folder *solutions*.
- Find the optimal strategy for this case.

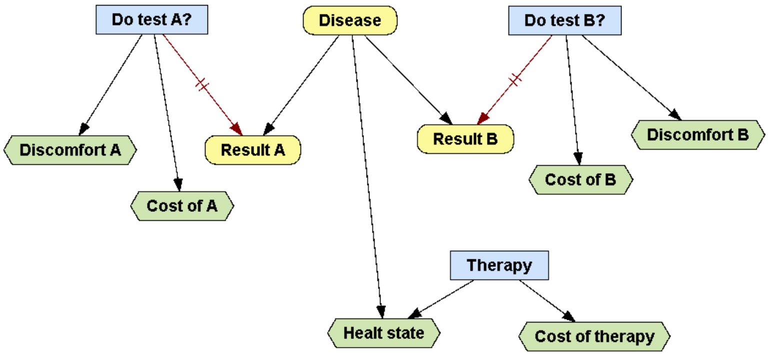
4. Cost-effectiveness analysis (atemporal)

Question

- Given the disease and the tests described in the previous exercise, what is the optimal strategy for each value of the willingness to pay?

Hints

- Depart from the network *DAN-two-tests-without-costs.pgmx* built in the previous exercise.
- Add the nodes *Cost A*, *Cost B*, and *Cost therapy*, as shown below. Check that the criterion is *Cost ($)* for each of them. Draw the links required and edit the corresponding utility tables. (Note that, even though discomfort was introduced as a negative effectiveness, all costs are positive.)



- Save the network as *DAN-two-tests-with-costs.pgmx*.
- Click the button 📈 to do a cost-effectiveness analysis. Answer the above question.

5. Cost-effectiveness analysis with a Markov model

Exercise

- A disease D may be latent or active. When it is latent, the probability of becoming active in the next month is 11% and that of dying is 2%. When the disease is active, the monthly probability of dying is 15%. Quality of life is 0.9 for latent and 0.7 for active. The cost of the standard therapy is £150/month when the disease is latent and £2,500/month when it is active.
- There is a new therapy, with a cost of £950/month, which slows down the progression of the disease, so that the monthly probability of becoming active reduces to 8%—without affecting the probability of death. Unfortunately, this therapy has no effect when the disease is already active.

- With an annual discount rate of 3.5% for both cost and effectiveness, what is the ICER of the new therapy?
- Is it cost-effective for a willingness-to-pay of £20,000 per QALY?

Hints

Check that the model offered in the solutions is correct

- Open the network *MID-latent-active.pgmx*, available in exercises.zip, in the folder *solutions*.
- Right-click on an empty zone of the network panel to open the Network properties dialog. Check that Netowrk type is *Markov influence diagram (MID)*.
- In the Network properties dialog, go to tab Advanced, click the button Decision criteria, and check that *cost* is measured in *British pounds* and *effectiveness* in *QALYs*. Go back to the Advanced tab, click Temporal options and check that the cycle length is *one month*.
- Open the Node properties dialog for the chance node *State [0]*. Observe that name is *State* and time slice is *0*. In the Domain tab, observe that the states are *latent*, *active*, and *dead*.
- Open the Node properties dialog for the decision node *Therapy*.
- Pay attention now to node *Therapy*, which represents the decision of whether applying the new therapy when the disease is latent. (When the disease is already active, the therapy brings no benefit, so we do not need to decide whether it should be applied or not.) This decision is made once and forever, so it is atemporal. Check it by opening the Node properties dialog for this node and observing that time slice is *atemporal*. In the Domain tab check that the options for this decision are *new* and *standard*.
- Open the Node properties dialog for *Cost of therapy [0]* and check that its criterion is *Cost (£)*.
- Similarly, check that the criterion for *Quality of life [0]* is *Effectiveness (QALY)*.
- Check that the links are correct. For example, the links pointing at *State [1]* mean that the patient's state in one cycle depends on the state in the previous cycle and (if the patient was in state latent) on whether the therapy is applied. There is no link from *Therapy* to *Quality of life [0]* because the therapy has no direct effect on the quality of life: its acts indirectly by delaying the transition from latent to active.
- Open the probability table for *State [0]*. Check that initially all patients are in state *latent* because this is the subpopulation of interest (when the disease is active or the patient is dead, it is not necessary to do a cost-effectiveness analysis to decide that the therapy should not be applied).
- Open the probability table for *State [1]* and check that the transition probabilities agree with the statement of the problem.
- Open the utility table for *Cost of therapy [0]* and check that it is correct. Do the same for *Quality of life [0]*.

Temporal evolution of the variables

- Right-clicking on *State[0]* or *State [1]* and select Temporal evolution. Set the horizon to *60 cycles*, i.e., 5 years, and accept the default value One decision. You will obtain 6 curves showing the probability of each state evolves depending on the therapy applied.

  If you wish to know the probability of being alive in a cycle (i.e., the proportion of patients that are alive), uncheck the box for *dead* and click the radio button Sum; this way OpenMarkov will show the sum of the probabilities of *latent* and *active*. The new therapy clearly increases life expectancy. If you wish to know the probability of the disease being *active* in each cycle, also uncheck the box for *latent*. You will see that the new therapy make patients transition more slowly to state *active*.

- Observe the temporal evolution of *Cost of therapy*. This cost decreases as patients transition from *latent* (the only state in which the new therapy is applied) to the states *active* or *dead*. Click the Cumulative radio button to obtain the accrued cost.
- Observe the temporal evolution of *Quality of life*. The Cumulative radio button allows us to observe its value accrued over time, i.e., the effectiveness.

Cost-effectiveness analysis

- Click the button 📈 to do the cost-effectiveness analysis. Each discount can be set per year or per cycle. In accordance with the statement of the problem, they are set to *3.5% per year*. OpenMarkov will transform them into per-cycle discounts.
- In the Scope selector, accept the default value, *One decision*. The Analysis tab shows the cost and effectiveness of each intervention. The CE plane tab shows them on a plot. You can click the Relative to button to show the incremental cost and effectiveness. The Forntier interventions tab shows these values in a table, as well as the ICER we were looking for.