Information and coding theory

Project : part II

Deadline: 22th October

Goal

This part aims to help you to become familiar with graphical probabilistic models. Two aspects will be addressed in this project: the structure learning from a sample set of infinite size and the link between graphical models and independence relationships.

The structure learning will require a good understanding of learning algorithms and graphs representation.

A bit of theory

Graphs representation

There are several ways to represent a graph. We will use, in this project, the adjacency matrix. This method encodes a graph defined on n nodes by a square matrix of size n. In this work, we will never use a multi-arc, i.e. there will never be more than one arc between two given nodes.

In that case, for each pair of nodes i and j and an adjacency matrix A, we have:

- $A_{ij} = 0$ and $A_{ji} = 0$: no arc between i and j
- $A_{ij} = 1$ and $A_{ji} = 0$: directed arc from i to j
- $A_{ij} = 1$ and $A_{ji} = 1$: undirected arc between i and j

The diagonal elements indicate the presence (= 1) or the absence (= 0) of a loop connecting the node with itself.

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

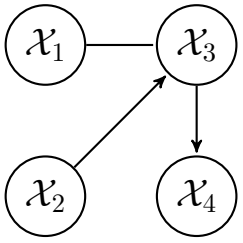

Figure 1: A graph defined on four variables and its adjacency matrix.

Algorithm of polytree recovery

When a distribution admits a polytree as perfect map it is possible to retrieve this polytree from a sufficiently large data set. The algorithm follows four steps:

1. Calculation of the mutual information two by two
2. Construction of a maximum (weighted) spanning tree
3. Orientation of arcs
4. Parameters setting

The arc orientation is based on two measures: the mutual information and the conditional mutual information. Variables are scanned one after another, starting from leaves towards the root, i.e. the center of tree. For each one, the idea is to detect if it is the summit of one (or more) v-structure.
A variable is the summit of a v-structure if two conditions are fulfilled:

i. The mutual information between its parents is equal to zero.

ii. The mutual information between its parents conditionally to the child is strictly greater than zero.

When a v-structure is detected, the process is stopped and the orientation information is propagated in the structure.

Practical informations

From now, the toolbox BNT[1] will be necessary and in particular the section "*Creating your first Bayes net*" from the user guide. As a reminder, try *test_BNT* in order to verify that you have successfully installed the toolbox.

You may find interesting the following functions: **graphminspantree**, **spy**, **biography**, **chi2pdf** or **squeeze**.

The script **code_part2** is necessary to use the function **ask_oracle(**A**,**B**,**C**)** which returns the mutual information between two variables, A and B, conditionally with the variable C (eventually empty) given by an oracle based on a infinite data set.

You have the function **order_node** which takes, as entries, an undirected tree structure and returns the node numbers ordered according to a path from the outside (from the leaves) towards the inside of the tree. Be careful, this function calls the function **neighbours**.

Finally, you also have the **UndirectedMaximumSpanningTree** function which returns the maximum spanning tree from a cost matrix. You can also use another method, in that case, you may find interesting the two following packages:

- http://www.mathworks.in/matlabcentral/fileexchange/13457,
- http://www.mathworks.in/matlabcentral/fileexchange/24327.

[1] https://code.google.com/p/bnt/

The report (**of maximum 5 pages**, in **french** or in **english**) and codes must be sent by mail to asutera@montefiore.ulg.ac.be. All your files (report **and** codes) **must be gathered in one archive** (format zip or tar). The mail subject and the archive name have to be of the following form: *[Coding] project - part2 - LAST_ NAME First_ name*[2].

Questions

1. With the help of the toolbox BNT, take back the graph of FIGURE 1, delete the undirected arc and create a bayesian network based on the resulting graph (using binary variables).

2. Once again, take back the graph of FIGURE 1 and orientate the undirected arc in all possible ways. Give the independences of both graphs and explain the effect of the orientation on the independence set.

3. Write a function **result = neighbours(graph,node)** which returns a vector with all node numbers that can be reached in a single step in the graph from the given node number.

4. Write a function **result = tree_skel()** which returns the adjacency matrix corresponding to the undirected skeleton of the optimal tree according to the data (in the sense of the Kullback-Leibler divergence).

5. Write a function **result = node_parent(candidates,node)** which returns the candidates which have been identified as parents of the given node based on the data and according to the v-structure detection method.

6. Write a function **result = tree()** which returns the adjacency matrix corresponding to the directed skeleton of the optimal polytree.

7. Compare results of functions **tree_skel** and **tree**. Which are the differences in terms of represented independences. One of these two networks is a *perfect map* of the distribution, which one?

[2]For example, for me, it would be [Coding] project - part2 - SUTERA Antonio.