Information and coding theory

Project : part III

Deadline: 10th November

Goal

This third part of the project aims to help you to become familiar with structure learning from a finite data set. You will also use the BNT toolbox in order to learn parameters and make probabilistic inference.

Mainly, you will have to use appropriately the given functions and interpret wisely your results.

A bit of theory

Mutual information distribution of two independent variables

When two random variables $\mathcal{X}$ and $\mathcal{Y}$ are independent, their mutual information is equal to zero. However, when this information is measured from a finite data set of size N, we only have an estimate of the true value. In the case of two independent variables, we have the following property :

$$2 * N * ln(2).\hat{I}(\mathcal{X}, \mathcal{Y}|\mathcal{Z}) \text{ converges asymptotically towards a } \chi^2 \text{ law}$$
$$\text{of degree of freedom} k = (C_{\mathcal{X}} - 1) * (C_{\mathcal{Y}} - 1) * C_{\mathcal{Z}},$$

where

- $C_{\mathcal{X}}$ is the cardinality of $\mathcal{X}$,
- $C_{\emptyset} = 1$.

We can thus use the χ^2 test to evaluate the independency hypothesis.

Probabilistic inference

The bayesian network exploitation allows to make predictions, derive probabilities of occurrence, ... There are several algorithms to make an inference based on the bayesian network and, for instance, the junction tree and the variable elimination.

Junction tree

The junction tree method is an inference technique which involves two parts. The first one consists in transforming (with respect to some conditions, not described here) the bayesian network into a junction tree. The idea is to dissimulate each existing cycle into one or more nodes of the junction tree. The second step of the algorithm consists in making the inference calculation on the tree, by using reasonings similar to those of the

inference on a tree-bayesian network (for more information, see section 4.2 of Pearl[1]), which is very effective.

A (non-unique) junction tree of a graph $G = (V, E)$ is a pair (X, T), where $X = \{ X_1, \ldots, X_n\}$ is a family of subsets of summit V. We associate to each X_i a node of the tree T, such as:

- $\bigcup X_i = V$
- $\forall (v_i, v_j) \in E, \exists X_k$ such as $v_i, v_j \in X_k$
- If a summit $v \in V$ belongs to X_i and X_j, then $v \in X_k$ for all k such as X_k belongs to the (unique) path between X_i and X_j in T.

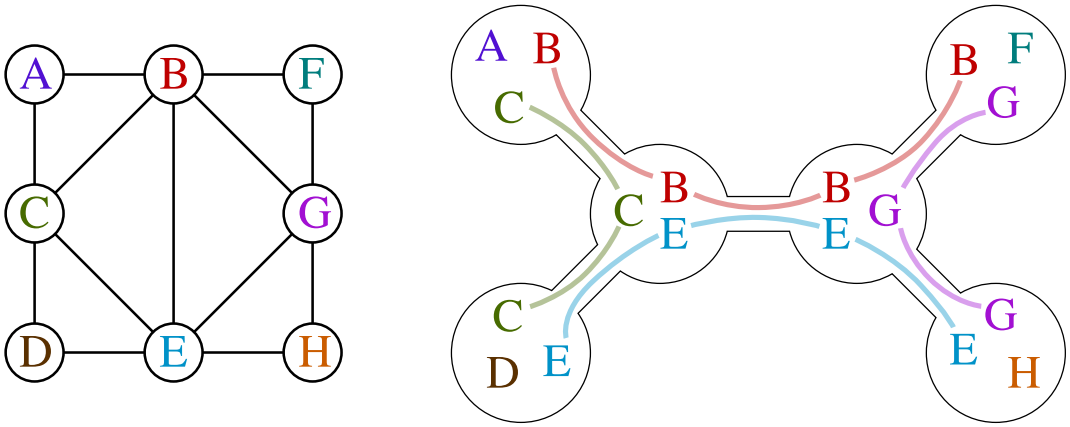

Figure 1: A graph and its junction tree.

Variable elimination

This method performs the information propagation by decreasing progressively the size of the network by successive marginalizations. Therefore, the number of variables decreases little by little, until there are only variables of interest.

When a variable is eliminated, the probability densities of nearby nodes (i.e. neighbours) are modified to take into account the disappearance. For example, let consider a Markov chain of three variables A, B, C such as $P(A, B, C) = P(A|B)P(B|C)P(C)$. If we want to calculate $P(A)$, we can directly marginalize by summing on B and C, but we can also decompose the operation in several steps by eliminating variables. Indeed, we can first remove the variable C, which does only modify the probability of B : $P(A, B) = P(A|B)P(B)$. Then, we can remove the variable B, to get $P(A)$.

Marginalize variables successively often reduces the number of required operations.

[1] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*

Practical informations

For this part, the toolbox BNT[2] will be also necessary and in particular the section *"Inference"* and, to a lesser extent, *"parameter learning"* of the user guide.
You may find interesting the function **chi2pdf**.
The script **code_part3** initializes three variables:

- *big_LS* which is a sample matrix of size 17x5015 used as learning set for the bayesian network parameters.
- *small_LS* which is a matrix of size 4x17x600 containing four learning sets of 600 samples.
- *dag* which is a adjacency matrix encoding a directed acyclic graph of seventeen nodes.

You also have two functions **bn_mut_inf** which calculates the conditional mutual information from a marginal distribution *marg*, as given by a inference engine of the BNT toolbox and **d_samples** to generate random probability tables.

The report (**of maximum 10 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 - part3 - LAST_NAME First_name*[3].

Questions

1. Create a bayesian network of two independent variables, and repeat a thousand times the following experiment: give random parameters to the network, and generate 50 samples. Calculate the mutual information between both variables and make a histogram. Is the obtained distribution consistent with the theory?
   NB1: Use ***CPT = d_samples(ones(1,2),1)*** *to generate randomly a probability table for a binary variable.*
   NB2: Use ***cell2num*** *to transform the generated samples into the usual matrix form.*

2. Write three functions **result = tree_skel(samples)**, **result = node_parent(samples,candidates,nodes)** and **result = tree(samples)**, which are the function of the previous project except that they work from a finite data set (*samples*). Explain.

3. Apply the function **tree(samples)** to the data sets of size 600. Are the results the same? These directed acyclic graphs, are they *I-map*, *d-map* or *perfect map* of the real distribution which has the structure *dag*?
   NB: Use ***squeeze*** *to remove a dimension of a matrix.*

[2] https://code.google.com/p/bnt
[3] For example, for me, it would be [Coding] project - part3 - SUTERA Antonio.

4. Create a bayesian network corresponding to the graph *dag*.

5. Use the *big_LS* data set to learn the parameters of this bayesian network to the maximum likelihood.

6. Calculate the marginal distribution of the disease variable. Which is the most frequent pathology according to this model?
   NB: Use ***var_elim_inf_engine*** *instead of* ***jtree_inf_engine*** *as inference engine.*

7. Calculate the joint marginal distribution of variables *iron* and *obesity*. Does the order in which you give the arguments matter?

8. Calculate the marginal distribution of the marginal distribution of the *disease* variable when the patient is a woman. Does it increase the risk of a pathology in particular?

9. Calculate the marginal distribution of the marginal distribution of the *disease* variable conditionally with the case of an abnormal GGTP rate. What could you deduce?

10. How evolve the probability distribution of the *disease* variable, if, in addition to the previous case, the patient has antimitochondrial antibody? Based on that result, which of both pathologies seems, according to you, linked with that kind of antibody?

11. Calculate the mutual information between variables 4 and 5 and between variables 16 and 17, from the bayesian network and from the data. How do you explain the results?

12. In the sake of economy, we can to reduce to number of tests given to a patient. Choose two variables (except *disease*) which, **together**, give the most accurate diagnosis. Explain.

13. In a second medical team, we also want to do two tests but sequentially: one after the other. If the first test is made on the variable which is most informative about the *disease*, what will be your second test(s)?