

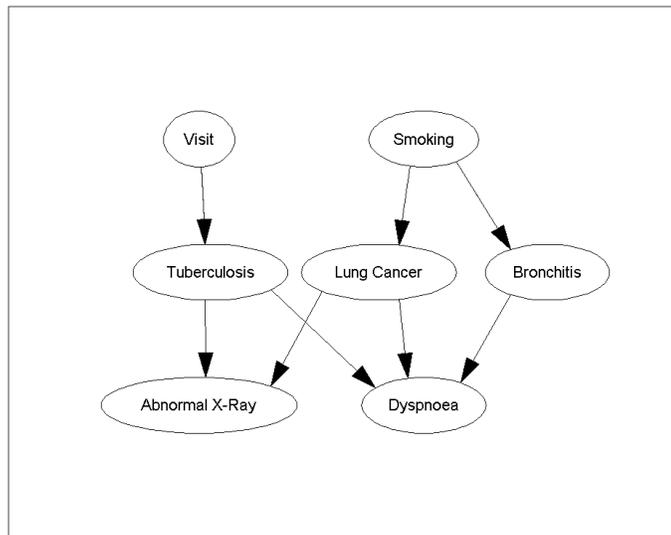
CPSC 540 Assignment 4 (due March 8)

DAGs and Project Proposal

1 D-Separation

Directed acyclic graphical (DAG) models are a framework for understanding many of the independence and computational issues associated with multivariate density modelling. In the case of independence, the d-separation rules allow us to answer whether conditional independencies between arbitrary sets of variables are implied by the structure of DAG model. This question will give you practice answering conditional independence queries using the graph structure alone.

In particular, consider the DAG model below, for distinguishing between different causes of shortness-of-breath (dyspnoea) and the causes of an abnormal lung x-ray, while modelling potential causes of these diseases too (whether the person is a smoker or had a ‘visit’ to a country with a high degree to tuberculosis).



We'll assume that the distribution over the variables is “faithful” to the graph, meaning that variables are conditionally independent iff the graph structure implies their independence. Under this assumption, [say why each of the following conditional independence statements is true or false:](#)

1. (Smoking) \perp (Bronchitis).
2. (Smoking) \perp (Dyspnoea).
3. (Tuberculosis) \perp (Lung Cancer).

4. (Abnormal X-Ray) \perp (Dyspnoea).
5. (Abnormal X-Ray) \perp (Visit) | (Tuberculosis).
6. (Bronchitis) \perp (Lung Cancer) | (Smoking).
7. (Tuberculosis) \perp (Lung Cancer) | (Dyspnoea).
8. (Visit,Smoking) \perp (Abnormal X-Ray, Dyspnoea) | (Tuberculosis, Lung Cancer, Bronchitis)
9. (Smoking) \perp (Visit) | (Dyspnoea).
10. (Tuberculosis) \perp (Bronchitis) | (Abnormal X-Ray).

2 Exact Inference

While DAGs can be used as a visual representation of independence assumptions, they can also be used to simplify computations. Before moving on to automated methods, this question will give you practice using the basic properties which allow efficient computations in graphical models.

For this question, let's assume that we use the following parameterization of the network:

$$\begin{aligned}
 & \text{Visit} \\
 & p(V = 1) = 0.01 \\
 & \text{Smoking} \\
 & p(S = 1) = 0.2 \\
 & \text{Tuberculosis} \\
 & p(T = 1|V = 1) = 0.05 \\
 & p(T = 1|V = 0) = 0.01 \\
 & \text{Lung Cancer} \\
 & p(L = 1|S = 1) = 0.10 \\
 & p(L = 1|S = 0) = 0.01 \\
 & \text{Bronchitis} \\
 & p(B = 1|S = 1) = 0.60 \\
 & p(B = 1|S = 0) = 0.30 \\
 & \text{Abnormal X-Ray} \\
 & p(X = 1|T = 1, L = 1) = 1.00 \\
 & p(X = 1|T = 1, L = 0) = 0.98 \\
 & p(X = 1|T = 0, L = 1) = 0.9 \\
 & p(X = 1|T = 0, L = 0) = 0.05
 \end{aligned}$$

Dyspnoea

$$p(D = 1|T = 1, L = 1, B = 1) = 0.90$$

$$p(D = 1|T = 1, L = 1, B = 0) = 0.70$$

$$p(D = 1|T = 1, L = 0, B = 1) = 0.85$$

$$p(D = 1|T = 1, L = 0, B = 0) = 0.65$$

$$p(D = 1|T = 0, L = 1, B = 1) = 0.82$$

$$p(D = 1|T = 0, L = 1, B = 0) = 0.60$$

$$p(D = 1|T = 0, L = 0, B = 1) = 0.80$$

$$p(D = 1|T = 0, L = 0, B = 0) = 0.10$$

Compute the following quantities (hints are given on the right, and these will be easier to do in order and if you use conditional independence properties to simplify the calculations):

0. $p(S = 1)$ (marginal of root node; can read from table)
1. $p(S = 0)$ (negation of marginal of root node; use sum to one constraint)
2. $p(L = 1|S = 1)$ (conditional of child node given parents; can be read from table)
3. $p(L = 1)$ (marginal of child node; marginalize over parent)
4. $p(X = 1|T = 1, L = 1)$ (conditional of child given parents; can be read from table)
5. $p(X = 1|T = 1)$ (conditional of child with missing parent; marginalize over missing parent)
6. $p(X = 1|T = 1, S = 1)$ (conditional of child given parent and grand-parent, marginalize over missing parent)
7. $p(X = 1)$ (marginal of leaf node; marginalize over parents and use independence to simplify)
8. $p(T = 1|X = 1)$ (conditional of parent given child; use Bayes rule)
9. $p(T = 1|L = 1)$ (conditional of parent given co-parent; use independence and then marginal)
10. $p(T = 1|X = 1, L = 1)$ (conditional of parent given child and co-parent; use Bayes rule)

3 Viterbi Decoding

The function *viterbiDemo.m* loads the initial state probabilities and transition probabilities for three Markov chain models on T binary variables,

$$p(x_1, x_2, \dots, x_d) = p(x_1) \prod_{j=2}^d p(x_j|x_{j-1}).$$

It then tries to find the optimal decoding (the most likely assignment to the variables $\{x_1, x_2, \dots, x_d\}$) in each of the three chains. In the demo, decoding is done by enumerating all possible assignments to the variables. This works for the first two chains as they only have 4 variables, but is too slow on the last chain because it has 30 variables.

1.

Write a function, *exampe_viteribi*, that implements the Viterbi decoding algorithm for Markov chains. You should use a 2 by d matrix V to represent the dynamic programming table, and another matrix T containing the argmax that lead to each entry in the table. [Hand in this code and report the optimal decoding of the third Markov chain.](#)

2.

Write a function, *marginalDecode.m*, that returns the sequence of states x_j that maximize the marginal probability $p(x_j)$ (for each j). [Hand in this code and report the sequence of most likely states.](#)

4 Project Proposal

For the final of this assignment, you must [submit a project proposal](#) for your course project. The proposal should be a maximum of 2 pages (and 1 page or half of a page is ok if you can describe your plan concisely). The proposal should be written for me, so you don't need to introduce any ML background but you will need to introduce non-ML topics. The projects must be done in groups of 2-3. If you are doing your assignment in a group that is different from your project group, only 1 group member should include the proposal as part of their submission (we'll do the merge across assignments, and this means that assignments could have multiple proposals). Please state clearly who is involved with each project proposal.

There is quite a bit of flexibility in terms of the type of project you do, as I believe there are many ways that people can make valuable contributions to research. However, note that final deliverable for the project will be a document containing at most 6 pages of text (the actual document can be longer due to figures, tables, references, and proofs) that emphasizes a particular "contribution" (i.e., what doing the project has added to the world). The reason for this, even though it's strange for some possible projects, is that this is the standard way that results are communicated to the research community.

The three main ingredients of the project proposal are:

1. What problem you are focusing on.
2. What you plan to do.
3. What will be the "contribution".

Also, note that for the course project that negative results (i.e., we tried something that we thought we would work in a particular setting but it didn't work) are acceptable.

Here are some standard project "templates" that you might want to follow:

- **Application bake-off:** you pick a specific application (from your research, personal interests, or maybe from Kaggle) or a small number of related applications, and try out a bunch of techniques (e.g., random forests vs. logistic regression vs. generative models). In this case, the contribution would be showing that some methods work better than others for this specific application (or your contribution could be that everything works equally well/badly).
- **New application:** you pick an application where ML methods have previously applied, and you test out whether ML methods are effective for the task. In this case, the contribution would be knowing whether ML is suitable for the task.
- **Scaling up:** you pick a specific machine learning technique, and you try to figure out how to make it run faster or on larger datasets (for example, how do we apply kernel methods when n is very large). In this case, the contribution would be the new technique and an evaluation of its performance.
- **Improving performance:** you pick a specific machine learning technique, and try to extend it in some way to improve its performance (for example, how can we efficiently use non-linearity within graphical models). In this case, the contribution would be the new technique and an evaluation of its performance.

- **Generalization to new setting:** you pick a specific machine learning technique, and try to extend it to a new setting (for example, making a graphical-model version of random forests). In this case, the contribution would be the new technique and an evaluation of its performance.
- **Literature review:** you pick a specific topic in ML, read a larger number of papers on the topic, then write a report summarizing what has been done on the topic. In this case, the contribution would be your summary of the relationships between the existing works.
- **Coding project:** you pick a specific method or set of methods (like independent component analysis), and build an implementation of them. In this case, the contribution would be the implementation itself.
- **Theory:** you pick a theoretical topic (like the variance of cross-validation), read what has been done about it, and try to prove a new result (usually by relaxing existing assumptions or adding new assumptions). The contribution could be a new analysis of an existing method, or why some approaches to analyzing the method will not work.

The above are just suggestions, and many projects will mix several of these templates together, but if you are having trouble getting then it's best to stick with one of the above templates. Also note that the above includes topics not covered in the course (like random forests), so there is flexibility in the topic, but the topic should be closely-related to ML.

This question is mandatory but will not be formally marked: it's just a sanity check and an excuse for you to allocate some time to thinking about the project.