SRI, 8 Feb 2008

Bayesian Belief Nets:
Demo and Introduction to Hugin

John Rushby

Computer Science Laboratory
SRI International
Menlo Park CA USA

Overview

- Background and motivation
- Example 1: multi-legged assurance cases
- Example 2: car crash
- Example 3 (develop the model, GUI details): jury fallacy

Background and Motivation

- Suppose we have test and verification results for a system and want to use these to certify it
- We want to be sure the system is good, i.e., its probability of being correct is very close to 1
- To talk about it being correct, we need specification
- And to test it, we need an oracle
- These also have some probability of being correct
- And there will be relationships among them
- E.g., P(oracle is correct) surely depends on P(specification is correct)
- I.e., the conditional probabilities P(oracle correct $|$ spec correct) and P(oracle correct $|\neg$spec correct) are of interest

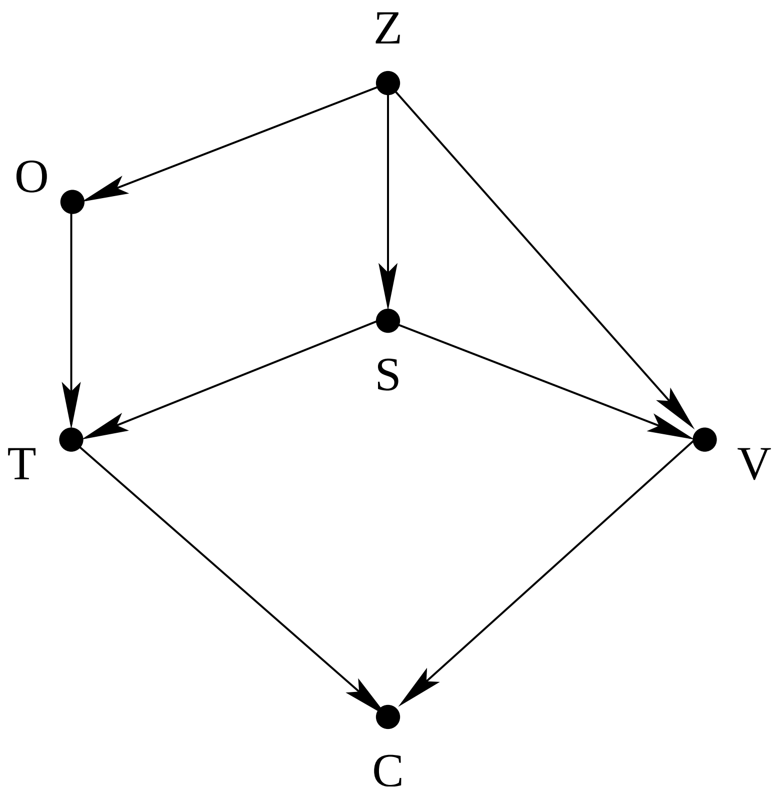
Bayesian Models

- We can use experience and expert judgement to propose values for these conditional probabilities
  - This is a model in this context
- Most natural to use subjective (i.e., Bayesian) interpretation of probabilities
- Then we can feed in known or assumed values for some of the individual probabilities
  - E.g., we know the test results
- And let them ripple through
- It's easy to ripple forwards through the conditional probabilities
  - $P(B) = P(B|A) \times P(A) + P(B|\neg A) \times P(\neg A)$
- To go backward, we use Bayes' rule
  - $P(A|B) = P(B|A) \times P(A)/P(B)$

Bayesian Belief Nets (BBNs)

- Now, we have several variables in our model, so we will have complex conditional probabilities like $P(A|B \wedge C|\neg D \vee E)$
- It is really hard to do Bayes rule over large collections of terms like this
- We simplify things if we can state what variables are unrelated
- A Bayesian Belief Net (BBN) is a graphical way to do this
- Just indicate the direct relationships as a graph

A BBN Example



Z: System Specification

O: Test Oracle

S: System's true quality

T: Test results

V: Verification outcome

C: Certification decision

BBN Tools

- A BBN model is a graph, plus conditional probability tables for each variable in terms of its direct ancestors
  - E.g., $P(O|Z) = 0.999$, $P(O|\neg Z) = 0.05$
- A BBN tool gives us a GUI to enter these, and a computational engine that lets us do "what if" experiments, like a spreadsheet
- My understanding is that there was some breakthrough a decade or so ago that made the computations feasible
- Hugin is one such tool, Hugin-Lite is the free version (models are limited in size)
- So let's try it

Multi-Legged Assurance Cases

- Littlewood and Wright analyzed this example analytically
- More sophisticated interpretation of some of the variables
    - Testing delivers X% confidence system is Y% correct
- Found paradoxical results for some versions of the model
    - E.g., more test success, less system correctness
    - Because it raises doubts about the test oracle
- They showed these paradoxes disappear when one of the legs has the characteristic of (idealized) verification
    - I.e., $Y = 100$ (perfection of the system)
    - But the verification itself could still be flawed
- My interest: get a numerical feel for these issues, esp. where verification is against a weak spec (e.g., static analysis)
- And in feasibility of BBNs for real certifications

Feasibility for Real: Car Crash Example

- Single car accident, hit a tree at 3am (in Holland)
- The female driver was sitting on the ground, next to the car, and stated three times that “he” had pulled the handbrake
- A badly injured male passenger was sitting on the front passenger seat
- The handbrake was in pulled position
- The car had been driven through a curve in the road right before it crashed
- There were tire marks from locked wheels in the curve of the road
- There were tire marks from a skidding car; the marks led to the place of the accident
- Neither driver nor passenger could remember anything

Car Crash Example

- Under Dutch law, the driver is assumed responsible in a single-car accident
- But this one was challenged in court
- Driver said passenger caused accident by pulling handbrake
- Passenger said driver caused it by speeding
- Analyzed in Hugin by P. E. M. Huygen (Computer/Law Institute, Amsterdam)
- Quite widely cited
- I thought I'd type it in

Car Crash Example: Issues

- What I found
- Some of the probability tables make no sense
- Some of the entries are missing
- Cannot reproduce the quoted values
- Might just be a careless author
  - Plus, can experiment with different parameters
- But I have doubts about the actual model
- E.g,, the skidmarks that indicate locked wheels should be a child of locking, not speeding
- The more you look at it, the more different, plausible, ways there are for building the model
- There is a nuke in Korea whose certification used a BBN with 80 variables

On the Other Hand: Jury Fallacy

- The jury, in a serious crime case, has found the defendant not guilty
- It is subsequently revealed that the defendant had a previous conviction for a similar crime
- Does the subsequent evidence of a previous similar conviction make you less confident that the jury were correct in their verdict?
- Most people think it does

Jury Fallacy

- Just building a model raises valuable issues
- In particular, to get to trial, the defendant had to be charged
- The prosecutor's decision to press charges is surely influenced by their knowledge of previous convictions ("round up the usual suspects")
- This could be a determining factor
- BBNs allow us to explore it
- If anyone wants to learn how to operate Hugin in more detail, we can build a model for this example