Eliciting Bayesian Networks via Online Surveys:

A New Approach to Knowledge Elicitation

by

Peter Serwylo

Bachelor of Multimedia Systems (Programming)

Bachelor of Information Technology and Systems (Honours)

Thesis submitted in full requirement for the degree of Doctor of Philosophy

Faculty of Information Technology

Monash University

April 2016

“The only thing wiser than anybody is everybody”
Charles-Maurice de Talleyrand

Contents

List of Figures xv

List of Tables xviii

List of Algorithms xix

Nomenclature xxi

Declaration xxi

Acknowledgements xxv

Abstract 1

1. Introduction 3
   - 1.1. Motivation . . . 3
   - 1.2. Research Questions and Goals . . . 5
   - 1.3. Research Background . . . 7
     - 1.3.1. Bayesian Networks (BNs) . . . 9
     - 1.3.2. Knowledge Acquisition . . . 11
     - 1.3.3. Surveys . . . 12
     - 1.3.4. Summary of Previous Research . . . 13
   - 1.4. Research Methodology . . . 14
   - 1.5. Significance . . . 15
   - 1.6. Contributions . . . 16
     - 1.6.1. Contributions to Theory . . . 16
     - 1.6.2. Contributions to Practice . . . 17
     - 1.6.3. Contribution to Methodology . . . 17

1.7. Scope of the Research Project 18
1.8. Thesis Structure 19

2. Literature Review 21
2.1. Computer Models for Problem Solving 21
2.1.1. Constructing Computer Models 22
2.2. Bayesian Networks (BNs) 24
2.2.1. What is a BN? 24
2.2.2. Considerations When Choosing a BN in Preference to Other Models 28
2.2.3. Typical Ways in Which BNs are Constructed 29
2.2.4. Eliciting the Structure of a BN From Experts 33
2.2.5. Eliciting the Probabilities Required for a BN 36
2.2.6. Summary of BNs 41
2.3. Knowledge Acquisition 41
2.3.1. Relationship Between KA and Knowledge Engineering . . . 42
2.3.2. Techniques used in KA 42
2.3.3. Reliability and Validity in Expert Judgement 44
2.3.4. Biases During KA 46
2.3.5. Quantifying Expertise of Experts 47
2.4. Combining Expertise 49
2.4.1. Combining Experts Opinions During Elicitation 49
2.4.2. Statistical Techniques for Combining Results 50
2.5. Surveys 53
2.5.1. Total Survey Error 53
2.5.2. Surveys as a Knowledge Acquisition Tool 54
2.6. Chapter Summary 55

3. Methodology 57
3.1. The Design Science Research Methodology 57
3.2. Problem Identification 58

3.3. Propositions 60
3.3.1. Times When SEBN Would be More Suitable Than Traditional KEBN 60
3.3.2. Times When Traditional KEBN Would be More Suitable Than SEBN 64
3.3.3. Quality of BNs Elicited Using SEBN vs Traditional KEBN . 65
3.4. Artifacts Produced From This Research 67
3.4.1. Method to Elicit BNs Using Online Surveys 67
3.4.2. Open Source Web Survey Application 68
3.4.3. Note on the Terminology of “Instantiation” 68
3.5. Evaluation 69
3.5.1. Empirical Evaluation of SEBN 69
3.5.2. Iterative Evaluation During Research 71
3.6. Chapter Summary 72

4. Building BN Structure Through Survey Based Elicitation 73
4.1. Overview 73
4.2. Generating Questions 75
4.2.1. Initial Basis of All Questions 76
4.2.2. Making Survey Questions Easy to Understand 77
4.2.3. Manual Intervention and Rewording of Questions 78
4.2.4. Dealing with Questions that Don’t Make Sense 79
4.3. Constraining Possible Questions 80
4.4. Allocating a Subset of Questions 85
4.4.1. What is Allocated? 86
4.4.2. Question Allocation Strategies 86
4.4.3. How Many Allocated Questions is Enough? 87
4.5. Collating Answers 90
4.5.1. Note on Times When Multiple Models can be Created and Tested 92
4.6. Identifying and Dealing With Anomalies 92
4.6.1. Cycles in Network, Resulting in an Undirected or Cyclic Graph 93
4.6.2. Indirect vs Direct Relationships 95

4.6.3. Optimisations to Aid in Subsequent CPT Elicitation 97
4.7. Producing BN Structure 100
4.8. Chapter Summary 101

5. Evaluating Structure Elicitation 103
5.1. Experimental Method Used for Evaluation 104
5.1.1. Overview of Experimental Method 104
5.1.2. Note on the Iterative Nature of the Evaluation Survey . . . 105
5.1.3. Choice of Existing "Gold Standard" BN 105
5.1.4. Recruiting Participants for Survey 106
5.1.5. Software Used for Evaluation Survey 109
5.1.6. Duration of Evaluation Survey 110
5.2. Constructing the BN Structure From Survey Responses 111
5.2.1. Collating Responses into a Directed Graph 111
5.2.2. Removing Cycles From the Collated Graph 114
5.2.3. Summary Matrices 115
5.2.4. Number of Arcs in Network Structures 122
5.3. Scoring the Evaluation Networks Against the Gold Standard 122
5.3.1. Scoring Networks Using the SHD Metric 124
5.3.2. Scoring Networks Using the ROC and F1 Metric 127
5.3.3. Scoring Networks by Parameterising Then Using the BIC Metric 131
5.3.4. Intentionally Degrading Survey Response Data (Post-Hoc) . 136
5.3.5. Summary of "Good" Networks 136
5.4. Time Spent by Participants Doing Survey 138
5.5. Comparing Estimated Quality Measures to Accuracy 140
5.6. Optimising Network Structure for Future CPT Elicitation 141
5.6.1. Changes to *Eval* Networks to Facilitate Evaluation of Optimisations 142
5.6.2. Parameter Reduction When Applying NoisyMAX Optimisation 144
5.6.3. Parameter Reduction When Applying NoisyOR Optimisation 144

5.7. Limitations and Post-hoc Exploratory Analysis 145
5.7.1. The Cost of Excluding Arcs With Low Strength 145
5.7.2. Investigation of Arcs With High Strength 147
5.7.3. Summary of Post-hoc Analysis 151
5.8. Chapter Summary 152

6. Calculating BN Probabilities Through Survey Based Elicitation 155
6.1. Overview 155
6.2. Generate Questions 156
6.2.1. Questions Required for Variables With no Parents 157
6.2.2. Questions Required for Variables With a Single Parent . . . 160
6.2.3. Questions Required for Variables With Multiple Parents . . 162
6.3. Allocating a Subset of Questions 168
6.3.1. What is Allocated 168
6.3.2. Question Allocation Strategies 169
6.3.3. How Many Allocated Questions is Enough? 170
6.4. Collating Answers 171
6.5. Trade-offs 171
6.5.1. Preventing Duplicate Questions When Duplicate CPCs Arise 172
6.5.2. Randomising question order 172
6.6. Chapter Summary 173

7. Evaluating Probability Elicitation 175
7.1. Experimental Method Used for Evaluation 175
7.1.1. Overview of Experimental Method 176
7.1.2. Recruiting Participants for Survey 177
7.2. Comparing CPTs of Individual Variables 180
7.2.1. Collating Survey Responses into CPTs 180
7.2.2. Summary of Probabilities 182
7.3. Verifying Coherency Among Responses 184
7.4. Time Spent by Participants Doing Survey 185
7.5. Verifying Accuracy of Das Estimations 186
7.6. Chapter Summary 188

8. Conclusions and Future Work 191
8.1. Thesis Summary . . . 191
8.2. Revisiting the Propositions Post Evaluation . . . 195
8.2.1. Measuring Time Commitments (Proposition 1 & 2) . . . 195
8.2.2. Geographically Dispersed Participants (Proposition 3) . . . 198
8.2.3. Distinguish Level of Expertise (Proposition 4) . . . 199
8.2.4. Traditional Method More Flexible (Proposition 5) . . . 199
8.2.5. Traditional Method When Less Experts (Proposition 6) . . . 200
8.2.6. Not Identical, but Similar Structure and Score (Proposition 7, 8 & 9) . . . 202
8.3. Addressing the Research Questions and Goals . . . 203
8.3.1. Addressing the Sub-Questions . . . 203
8.3.2. Addressing the Main Research Question . . . 205
8.4. Reflections on Methodology . . . 206
8.4.1. Existing Threats to Instantiation Validity . . . 207
8.4.2. Additional Threats to Instantiation Validity . . . 208
8.4.3. Addressing Instantiation Validity . . . 211
8.5. Research Contributions . . . 211
8.5.1. Contributions to Theory . . . 212
8.5.2. Contribution to Practice . . . 212
8.5.3. Contributions to Methodology . . . 213
8.6. Limitations . . . 213
8.7. Future Work . . . 214
8.7.1. Further Evaluation with Experts . . . 214
8.7.2. Evaluate the BN Elicitator Software . . . 215
8.7.3. Further Investigation and Validation of the Weighted Sum Algorithm . . . 215
8.7.4. Variable Elicitation and Classification via Surveys . . . 216
8.7.5. Alternate Methods to Elicit BN Structures . . . 216
8.7.6. Investigate Different Collation Algorithms . . . 217
8.7.7. More Allocations . . . 217
8.7.8. Heuristics for Removing Cycles . . . 218
8.7.9. Modifying BNs . . . 219

8.7.10. Other Odd Looking and Non-Idiomatic Patterns or Optimisations . . . 219
8.7.11. Further Refinement of BNE . . . 220

Bibliography **221**

A. BNE Implementation Details **237**
A.1. Tech Stack . . . 237
A.2. Configuration . . . 238
A.3. Conducting SEBN via BNE . . . 239
A.4. Analysing Results . . . 240
A.5. Future work . . . 241
A.5.1. GUI Configuration . . . 241
A.5.2. Multiple surveys on one installation . . . 241
A.5.3. Allocating Variables Based on Number of Responses . . . 241

B. Results of Collating Responses **243**
B.1. Cycle Removal for Maj Network Structures . . . 243
B.2. Cycle Removal for DS Network Structures . . . 245

C. Results of Structure Evaluation **253**

D. Issues Surrounding Choice of Gold Standard BN **257**
D.1. Differences Between Gold Standard BNand Evaluation Network . . . 257
D.1.1. Variables Removed for Structure Elicitation . . . 257
D.1.2. Variable States Removed for CPT Elicitation . . . 258
D.1.3. Issues Arising due to Variable and State Removal . . . 259
D.2. CPTs Which Were Evaluated . . . 261
D.3. Other Issues with Gold Standard BN . . . 262

E. Alternative Variable Classes **265**

F. Question Generation Examples **269**
F.1. Structure Elicitation Questions . . . 269

F.2. Probability Elicitation Questions 270
F.2.1. Variables with Zero Parents 271
F.2.2. Variables with One Parent 271
F.2.3. Variables with Multiple Parents 272

List of Figures

1.1. High level overview of an example car insurance risk assessment system using a BN to help make decisions. 8
1.2. Example “Cancer” Bayesian network (from Cooper, 1999). 9

2.1. Example BN modelling the relationship between smoking, radiation, cancer and fatigue. 25
2.2. Example of causal arcs which also represent the correct conditional independence relationship between the three variables. 34
2.3. Example of parent divorcing. 34
2.4. Examples of two of the five idioms proposed by Neil et al. (2000). . 35
2.5. Four general classes of variable, and the logical dependencies between them proposed by Kjærulff and Madsen (2013, p152-154). 36
2.6. Example of a factorised PET requiring 50% of the probabilities than a CPT (adapted from Friedman and Goldszmidt, 1999, p4). 37
2.7. “The Fragment of Text and Probability Scale for the Assessment of the Conditional Probability” adapted from van der Gaag et al. (1999). 40
2.8. Visualization of reliability vs validity: The centre of the target is the “true” value being measured with imperfect tools, and each imperfect measurement is a black dot on the target. 45

3.1. As more experts participate in a survey, each is allocated less questions. Whereas with traditional KEBN, more experts in a group will likely increase the chance of disagreement which require resolution. 65
3.2. The two main artifacts produced as a result of this research. 67

3.3. A broad overview of the evaluation process, whereby a survey is conducted, and the resulting BN(s) are then compared to an existing, gold standard to see how closely they match. 70

4.1. Basic overview of the survey process. Each stage will be elaborated on in greater detail as this chapter progresses. 74

4.2. Procedural generation of questions for the survey. Note that although “classify variables” is conducted prior to “generate questions” in the flowchart, it makes more sense for the purpose of explaining the survey technique to discuss question generation first. 75

4.3. Two different ways to visualise the relationships between variables - as a network and as a matrix. 76

4.4. Before generating questions, variables are classified into categories. This enables the total number of questions to be reduced. 81

4.5. Four general classes of variable and the logical dependencies between, them proposed by Kjærulff and Madsen (2013, p152-154). 81

4.6. Illustration of the variable classification scheme proposed by Kjærulff and Madsen (2013). Note how the number of possible relationships in the matrix is greatly reduced, thus reducing the number of questions required by a survey. 82

4.7. Relaxing constraints on the variable classes, by allowing intra-class relationships. 83

4.8. After generating all of the survey questions, a subset of these are allocated to each expert so that no one person needs to answer every question. 85

4.9. Screenshot from BNE software configured with variables from the insurance network. 86

4.10. Visually exploring parameters for allocating questions to experts using the BNE software. 89

4.11. Collation of answers is the part of the process where multiple answers to the same question, each from different experts, are combined to decide on the “best” answer. 90

4.12. After collating answers, the structure is searched for anomalies to be resolved by the experts. 93
4.13. Three seemingly valid causal relationships that form a cycle when combined . In order to make a valid BN structure, one of the valid assumptions will have to be disregarded to break the cycle. 93
4.14. Example of how a legal DAG can end up with multiple cycles after introducing one relationship. 94
4.15. Example of a *direct*, *potentially redundant* relationship in the insurance network (Binder et al., 1997) *"Risk Aversion → Driving History"*. It is potentially redundant because the *indirect* mediating chain of *"Risk Aversion → Senior Training → Driving Skill→ Driving History"* may be a better way to explain the relationship. . 96
4.16. After resolving anomalies introduced by collating multiple responses to each question, a final BN structure is output, ready for use in popular BN software packages. 100
4.17. Summary of each of the stages in the process discussed in this chapter. 102

5.1. A broad overview of the process for conducting the evaluation survey and analysing the results. 104
5.2. Visualising the participation dropout between iterations. 109
5.3. Number of responses received about each variables questions for each iteration. 110
5.4. Graph showing how many arcs were removed in order to get a DAG, for each *Eval* network. 116
5.5. Summary matrix for the *Gold* network structure. Dark circles indicate the presence of an arc, and the darkness of the background colour indicates its strength. 117
5.6. Summary of the Maj_1 BN. Black dots indicate a relationship between the two variables, while the darkness of the cell indicates the strength of the relationship. 118
5.7. Summary of Maj_3 and $DS_{0.15}$ network structures. 120
5.8. These difference matrices show the Maj_3 and $DS_{0.15}$ arcs in black circles and the arcs from the *Gold* network in squares. 121

5.9. Number of arcs in the *Gold* and *Eval* networks (see Appendix C, p254, for full data). 122

5.10. Example of two network fragments with a SHD of three. Notice the addition of *Antilock* → *PropCost*, the removal of *Antilock* → *Accident*, and the reversal of *VehicleYear* → *Antilock*. 124

5.11. As a BN structure diverges more, the ability of it to fit a particular distribution degrades. The error bars show the minimum and maximum scores achieved over 100 randomly augmented networks with the same SHD value. 126

5.12. SHD of the *Eval* networks compared to the *Gold* network (see Appendix C, p255 for full data). 127

5.13. The ROC space of arcs. Those closer to the top left are better, because they are "more correct" (higher TPR) and "less incorrect" (lower FPR). 129

5.14. F1 score for the *Eval* networks when compared to the *Gold* network.130

5.15. Process for scoring elicited network structures when they do not yet have CPTs. Note that the data used in step 3 for learning CPTs, and that used for step 7 to score the evaluation network are both different data sets, randomly sampled from the same gold standard BN. 132

5.16. BIC score of various BNs, after being parameterised with progressively larger sample sizes. 133

5.17. Scores calculated for the *Gold* and *Eval* networks (see Appendix C, p253 for full data). 135

5.18. *Eval* network scores after removing between 0-30 random participants before collating structures. 137

5.19. Cropped view to better show *Eval* network scores after removing between 0-30 random participants before collating structures. 137

5.20. Analysis of time spent by participants using the survey software, who completed all questions allocated to them in the first round, shown as quartiles. 139

5.21. Process for evaluating whether SEBN is able to meaningfully discern the expertise of each participant or not. 140

5.22. The quality of participants responses as per the Ipeirotis et al. (2010) calculation compared to the actual accuracy measured from *Gold* vs $DS_{0.05}$ network. 141

5.23. Four ordinal variables from Maj_3 with ≥ 5 parents of which only one is not ordinal. The non ordinal parent is shown with a shaded background. 143

5.24. Theft variable from $DS_{0.10}$ with five parents of which three will be considered boolean for this analysis. The non boolean parents are shown with a shaded background. 143

5.25. Each point is an arc in the *Eval* network, that is either in the gold standard (true positive) or not (false positive). 146

6.1. Simplified flowchart showing the process for eliciting probabilities with surveys. It does not show anything particularly revolutionary, but will be expanded on in greater detail throughout this chapter. 156

6.2. Single variable with no parents, including the three possible states the variable can take. 158

6.3. Elicit the marginal probability of every state for each variable. . . . 158

6.4. “The Fragment of Text and Probability Scale for the Assessment of the Conditional Probability” adapted from van der Gaag et al. (1999).159

6.5. User interface of BNE showing a question for the *Client Age* variable, which has zero parents in the insurance network. Whenever a button is hovered over, the tool tip is exposed to clarify the meaning of the words from the van der Gaag et al. (1999) scale. 160

6.6. Child variable with a single parent. The states the variable can take are shown also. 161

6.7. Elicit the marginal probability of every state for each variable. . . . 161

6.8. Screenshot of BNE when asking about the *Liability Cost* variable, which has a single parent of *Accident*. 162

6.9. Single variable with multiple parents. The states that each parent and the child variable can take are shown also. 163

6.10. The process after adding the elicitation procedure for variables with multiple parents. The flowchart for multiple parents includes questions for both the weighted sum algorithm and also the AHP algorithm. 164

6.11. Eliciting the CPC from the three parents of *Car Value* shown in Figure 6.9: *Mileage*, *Vehicle Year* and *Car Type*. 165

6.12. Question to elicit the $\Pr(Car\,Value < \$1,000|$ $Mileage < 10k, Vehicle\,Age = current, Make\,Model = family\,sedan)$, where the three parent states represent the CPC when *Mileage* = *<10k*. 166

6.13. Questions required for a pairwise comparison between *Vehicle Age* and *Mileage*. The responses are used as input into the AHP process to determine relative weightings between all parent variables of *Car Value*. 167

6.14. The scale used to elicit the relative difference in influence that two different parents have on a child variable. The numbers adapted from Saaty (1990, p112) and the labels from Monti and Carenini (2000, p503). 167

7.1. A broad overview of the process used to evaluate the elicitation of CPTs via SEBN. 176

7.2. Visualising the participation dropout as the survey progressed. . . . 178

7.3. Number of participants who completed questions about each CPT. 178

7.4. Results of the demographic survey. Numbers in parenthesis are the proportion of participants who answered the question. 179

7.5. Comparison of the probabilities of the *Gold* network compared to the equivalent probabilities for the *Mean*, *DS*, and $Learnt_{Bayes}$ networks respectively. 183

7.6. Root Mean Squared Error of the conditional probabilities in $Mean$, *DS*, $Learnt_{MLE}$, and $Learnt_{Bayes}$ when measured against the probabilities in *Gold*. 184

7.7. Each point represents the sum of an individual participants elicited probability values for different states of a variable, conditioned on specific states of its parents. 185
7.8. Time spent by unique IP addresses participating in the online survey (though not necessarily finishing). Note that this is lower than the actual time spent by participants, due to the reasons discussed in Section 7.4. 186
7.9. Correlation between explicitly elicited conditional probabilities and their respective estimated value from the weighted sum algorithm (Das, 2004). 187

8.1. Exponential increase in estimated time required of each expert, when less experts are enlisted to elicit the structure of a BN. 201

C.1. Scores calculated for the *Gold* and *Eval* networks. 253
C.2. Number of arcs in the *Gold* and *Eval* networks. 254
C.3. SHD of the *Eval* networks compared to the *Gold* network. 255
C.4. F1 score for the *Eval* networks when compared to the *Gold* network.256

E.1. Comparison between generic and domain specific variable classes, and the inter-class dependencies they exhibit. 265
E.2. The network from Chan et al. (2010) after manually classifying variables into six different categories for the purpose of this analysis. . . 267

F.1. Screenshot from BNE software configured with variables from the insurance network. 270
F.2. The $CarValue$ variable has three parents, which means the CPT is elicited using the weighted sum algorithm when using SEBN. 273

List of Tables

3.1. Rough estimates of the relative time commitment required of the researcher for surveys and interviews. Estimates are indicative only, to show the occasions where surveys would be expected to take less of a researchers time. 62

5.1. Various different BN structures that will be referred to throughout this evaluation, and their meaning. The *Maj* and *DS* (aka *Survey*) structures are created from data obtained during evaluation, while the remainder are used in order to have something to compare these to. 113

5.2. Iterations required to turn the $DS_{0.05}$ graph into a DAG using the Margaritis and Thrun (1999) algorithm. Each of the 16 iterations resulted in the arc being reversed, not removed (See Appendix B for details of other *Survey* structures). 115

5.3. Summary of the evaluation metrics in Section 5.3, comparing *Eval* networks to the *Gold* network. Each *Eval* network is ranked from 1^{st} to 14^{th}. Metrics which were considered unacceptable relative to the other networks are greyed out. 138

5.4. Reduction in parameters required to parameterise random variables by using the NoisyMAX distribution. 144

5.5. Arcs with high level of agreement arising from the evaluation study in Chapter 5. 148

7.1. The different algorithms used to produce CPTs for evaluation. . . . 180

7.2. The variables which are evaluated in this chapter. 182

8.1. Summary of researchers time expected to be spent on traditional KEBN and SEBN. 196

B.1. Variables removed to eliminate cycles in the Maj_3 network structures. 243
B.2. Variables removed to eliminate cycles in the Maj_2 network structures. 244
B.3. Variables removed to eliminate cycles in the $DS_{0.001}$ network structure. 245
B.4. Variables removed to eliminate cycles in the $DS_{0.01,0.05}$ network structures. 246
B.5. Variables removed to eliminate cycles in the $DS_{0.10}$ network structure. 247
B.6. Variables removed to eliminate cycles in the $DS_{0.15}$ network structure. 248
B.7. Variables removed to eliminate cycles in the $DS_{0.20}$ network structure. 249
B.8. Variables removed to eliminate cycles in the $DS_{0.25}$ network structure. 250
B.9. Variables removed to eliminate cycles in the $DS_{0.30}$ network structure. 251
B.10. Variables removed to eliminate cycles in the $DS_{0.35}$ network structure. 252

C.1. BIC scores, calculated by sampling a data set from the *Gold* BN, to compare to *Eval* BNs. See Figure 5.15 (p132) for details on parameterisation. 253
C.2. Number of arcs in the *Gold* and *Eval* networks. 254
C.3. SHD of the *Eval* networks compared to the *Gold* network. 255
C.4. F1 score for the *Eval* networks when compared to the *Gold* network. 256

D.1. Ramifications of removing a single variable vs removing a single state. 260
D.2. The variables which were used in the evaluation in Chapter 7. . . . 264

E.1. Number of variables in each domain specific class, and the resulting number of survey questions required. 266

List of Algorithms

4.1. Basic algorithm to generate questions of the form "Does X influence Y?" 78

4.2. Adaptation of Algorithm 4.1 which takes into account manual rewording of questions of the form "Does X influence Y?" 80

4.3. Removing cycles from a directed graph, adapted from Step 4 & 5 of Margaritis and Thrun (1999, p4-5) 95

4.4. Search for all the *potentially redundant* relationships in a DAG. . . 97

6.1. Eliciting compatible parent configurations for a particular child variable. 163

Nomenclature

AHP	Analytic Hierarchy Process
AMT	Amazon Mechanical Turk
BN	Bayesian Network
BNE	BN Elicitator
CPC	Compatible Parent Configuration
CPT	Conditional Probability Table
DAG	Directed Acyclic Graph
DSR	Design Science Research
IS	Information Systems
ISDT	Information Systems Design Theory
KA	Knowledge Acquisition
KE	Knowledge Engineering
KEBN	Knowledge Engineering for Bayesian Networks
ROC	Receiver Operating Characteristic
SEBN	Survey Elicitation for Bayesian Networks
SHD	Structural Hamming Distance

Declaration

I hereby declare that this thesis contains no material which has been accepted for the award of any other degree or diploma at any university or equivalent institution and that, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text of the thesis.

The research for this thesis received the approval of the Monash University Human Research Ethics Committee (Reference number: CF12/1826 - 2012001011)

Peter Serwylo

20 October 2016

[redacted]

Acknowledgements

Laura, my awesome wife, coolest person I've ever met, this would never have been done without your continued support. This thesis has been around since we got our little Arthur man, then we got married, and finally we bought a house. If I was to take a guess, I'd say we are at least a tiny bit more like grown-ups then we were at the outset. Now it is time to let the thesis free and continue on with our amazing life together.

To Arthur: Meow. You were not very helpful, but you are extremely lovable. Thanks for your nocturnal company on the few late nights that I did spend on this thesis.

To my supervisors Grace and Frada. I am very grateful to you both for putting up with me and helping steer this thesis towards completion. I hope you are both proud of what it entails. Also, to all of the admin people in Caulfield who look after the students so well, you're all amazing.

Finally, to those who helped pass the time and make me continually want to come into uni. I will look back fondly on time spent drinking coffee with Mark and Sepehr. Chatting about open source and civil liberties with Greg. Trying and not being so successful at learning a few Vietnamese words from Hue. ~~Procrastinating~~ pomodoro-ing and creating tea puns with Liz. Lunches and Trivial Pursuit in the ever-dark DSSE kitchen with Dwi, Misita, Ariesta, Iwan, Pari, Maria, and whoever would join us. Chess with Percy. Briscola and maybe a few sneaky homebrews with Janis. Cat talk with Helen. Board games with Kevin, Tom, and the rest of the NICTA crew. The list goes on and on, and as I write this I am reminded again of all the great times. I hope you guys enjoyed all those little things as much as I did.

Abstract

Bayesian Networks (BNs) are popular computer models used to perform reasoning under uncertainty. They are popular as they are one of the few computer models able to be constructed by analysing historical data or expert elicitation. The main contribution of this thesis is a new workflow for expert elicitation of BNs, aiming to reduce the associated knowledge elicitation bottleneck. This bottleneck is caused by logistical issues associated with interviewing multiple experts, resolving conflicting opinions among experts, and the combinatorial explosion that occurs when eliciting probabilities. The workflow proposed in this thesis brings together research from the fields of BN construction, knowledge acquisition, the survey methodology, and crowd sourcing. The end result is a workflow for conducting online surveys (SEBN) which has been implemented in an open source online survey tool called BN Elicitator (BNE). This workflow allows a greater number of experts to contribute to the construction of BNs compared to traditional elicitation approaches. It also reduces the workload of the knowledge engineer facilitating the BN elicitation and minimizes the time required of each expert contributing knowledge to the BN.

Two evaluation surveys were conducted using the BNE software to measure how successfully the newly proposed technique was able to elicit BNs. The results of these evaluations showed a small improvement over the currently employed methods of eliciting BNs from experts (primarily face to face interviews) but also many undesirable outcomes. The improvements obtained allowed the BNE software to facilitate the elicitation of a BN was faster than would have otherwise been required for face to face interviews. However, the BN resulting from the evaluation did not compare favourably to an existing published BN. Issues with the methods used to collate survey responses into a BN were identified and discussed, such as

the choice of crowd sourcing algorithms. Such issues highlight the need for further research in this area to improve the accuracy of BNs elicited using online surveys.

1. Introduction

1.1. Motivation

Over the past decades, computer models have been used to help decision makers in a diverse range of fields such as business (e.g. wu Liao et al., 2008), medicine (e.g. Lucas et al., 2004, p205), and ecology (e.g. Pollino et al., 2007), amongst others. The ability to use models in place of potentially costly experiments or risky interventions saves time, money and effort for all involved. Despite the diverse range of model types available, there is one common trait they all share. The more accurately they represent the real world, the more useful they become for investigations and supporting decision makers. Thus, there is a continual drive from both industry and academia to find new and novel ways to build better, more accurate models, and to build them more efficiently.

This thesis focuses on a particular type of model called a Bayesian Network (BN, Pearl 1988). These have several desirable properties which contribute to their success. First and foremost, they are probabilistic in nature enabling them to explicitly deal with uncertainty. This allows them to perform reasoning even when there is only imperfect background knowledge available. Other desirable attributes include their graphical nature, which makes them intuitive to people who do not have a computer science background (Korb and Nicholson, 2011, p145), and their ability to represent complex relationships between many variables.

One of the main ways in which computer models, including BNs, are built is by using algorithms to analyse large historical data sets to identify patterns (e.g. Korb and Nicholson, 2011, p181-292). Continual advances in technology have allowed increasingly more data to be collected in many diverse fields. These same advances have also encouraged novel algorithms for processing data, making it more feasible to use data to build computer models.

The other way in which such models can be constructed is through Knowledge Acquisition (KA, e.g. Korb and Nicholson, 2011, p293-404). As its name suggests, KA is the process of eliciting knowledge that experts have acquired through education and experience. Experts have been shown to be helpful at transferring their knowledge and experience into models which can then be used to aid in decision making (Eriksson, 1992). BNs belong to a somewhat unique group of models which can be constructed by both data analysis and expert elicitation.

There are many text books on the topic of expert elicitation (e.g. Milton, 2008), including several that focus specifically on BN construction (e.g. Kjærulff and Madsen, 2013; Korb and Nicholson, 2011). In addition, there is a wealth of applied research using expert elicitation in the construction of BNs (e.g. Chan et al., 2010; Kuikka and Varis, 1997; Przytula and Thompson, 2000). However, there is also a well established understanding that KA results in a situation whereby the time, effort, and cost involved in acquiring the requisite knowledge is greater than the benefit gained by building a system using KA (d'Aquin et al., 2008, p21). This is commonly referred to as the knowledge elicitation bottleneck (Feigenbaum, 1977). Despite this interest from industry and researchers who opt to use KA as a tool for producing BNs, there is little research in developing new theories to elicit relevant expert knowledge and overcome the knowledge bottleneck. This is best summed up by Helsper and Van der Gaag (2002):

> "As more and more Bayesian networks are being developed for complex applications, their construction and maintenance calls for the use of tailor-made knowledge-engineering methodologies" (p680)

Areas which currently cause problems when eliciting BNs from experts include the sheer number of probabilities which need to be elicited (van der Gaag et al., 1999), the logistical issues of organising many experts together in order to conduct elicitation, and how to deal with conflicting opinions among experts during the elicitation process (e.g. Clemen and Winkler, 1999). This research aims to address these issues and provide a new and novel method of KA for BNs, making use of online surveys in place of more traditional interviews and focus groups. The following section elaborates on this goal in more detail.

1.2. Research Questions and Goals

The goal of this thesis was to propose a KA technique for constructing BNs which captures the expertise of a greater number of domain experts compared to existing approaches. When conducting expert elicitation, the goal is to document as accurately as possible the current expertise in a particular field. If only a small number of experts participate in a KA process, then it stands to reason that there is a higher chance of missing out on important expertise. Thus by increasing the number of experts participating in KA, the resulting model will be less biased towards opinions of the few experts who contribute by including a broader range of knowledge from a more diverse group of experts. The more experts contribute, the better the resulting model is likely to be. This will be achieved by replacing the face-to-face interview component of traditional KA, with online survey questionnaires, as they have the potential ability to reach a greater number of people.

Hence, the main research question for the thesis was:

> *How can the process of eliciting knowledge for construction of Bayesian Networks be improved by making use of online surveys instead of face-to-face interviews?*

This research started from the premise that KA has been shown as a viable and useful way to construct BNs (e.g. Chan et al., 2010; Hoverman et al., 2011; Martin et al., 2005). The project focused on one aspect of this process, the face-to-face interviews, and investigated how the knowledge bottle neck introduced by these interviews can be alleviated through the use of online surveys. Primarily, the research aimed to make it possible to include a wider and more diverse range of knowledgeable experts in the KA process. In order to facilitate more expert contributions, the following sub-questions were also be addressed:

1. *As more experts are consulted, how can the total time and effort involved in KA for BNs be reduced?*

2. *As more expert opinions are gathered, how can they be collated into a single BN model without significantly increasing the workload to resolve differences?*

By addressing these sub questions, this thesis produced a KA technique for eliciting BNs via online surveys. Compared to traditional face-to-face interviews, this new technique exhibits the following characteristics:

- *Less time* required of experts and researchers alike.
- *Easier integration of differing opinions* into the BN model.
- *Less constraints* on where and when experts can contribute.

Less Time If the time commitment is large, then experts may reconsider setting aside the time to contribute. Existing survey research often uses monetary payments or prizes to encourage participation (Bosnjak and Tuten, 2003), however even with additional incentives, lengthy processes still deter people from participating (Marcus et al., 2007). The less time required of experts, the more it will encourage them to commit their time to a KA process (Marcus et al., 2007). As such, this thesis focuses heavily on reducing the number of questions and subsequent amount of feedback that is required of each expert.

Easier Integration of Differing Opinions As more experts become involved in a KA process, there will be more opinions that need to be incorporated into the model. As the number of conflicting opinions increase, so does the difficulty of resolving those differences. This research addresses the problem by leveraging algorithms from the field of crowd sourcing to combine opinions from multiple people.

Less Constraints Finally, if experts need to be present at a specific location at a given time (e.g. for a focus group with other experts), it reduces the chance they will be available to participate. Removing this constraint should ensure more experts will be available to contribute to the model. Therefore this research proposed an online survey based technique which can be completed online at the time of the experts choosing.

The goal was to ensure that the magnitude of the elicitation task does not increase too much as more experts are included. The end result is that the number of potential experts who are able and willing to contribute their knowledge to a model is maximised. In addition, this research is intended to encourage others to investigate better ways of eliciting knowledge for BN construction.

1.3. Research Background

For a long time people have used models to hide or abstract away complexities, test predictions and generally provide a better understanding of the world. A model train hides the complexity of a steam engine from children; Newtons model of gravity allowed calculations of how the planets and other smaller objects move; and IBMs “Watson” artificial intelligence tries to model as much general purpose knowledge as possible so that people can query it effectively.

It is becoming more common for decision makers in fields such as business (e.g. wu Liao et al., 2008), medicine (e.g. van der Gaag et al., 2002), genetics (e.g. Friedman et al., 2000), ecology (e.g. Shenton et al., 2013), and others to enlist the help of computer models in order to aid in decision making. As technology advances, a broader range of areas have taken advantage of computer models with a higher degree of success.

To illustrate an example, imagine a system to help insurance workers estimate the risk of insuring particular drivers (Figure 1.1). As with most software, such systems can be quite complex. This thesis was not concerned with the user interface or the database, but rather the one or more underlying *models* which perform calculations, make predictions, or generally help in making decisions. The example system in Figure 1.1 opts to make use of a BN model. When using a BN model to help aid in decision making for an insurance broker, the model takes in as much evidence about a particular client as possible (e.g. age, number of previous accidents, etc). The BN will then provide estimations of any remaining variables which have not yet been specified as evidence, such as the chance of them having an accident, or their car getting stolen. In summary, the user specifies evidence about a client, and the model outputs the expected cost to the insurance company for insuring that client.

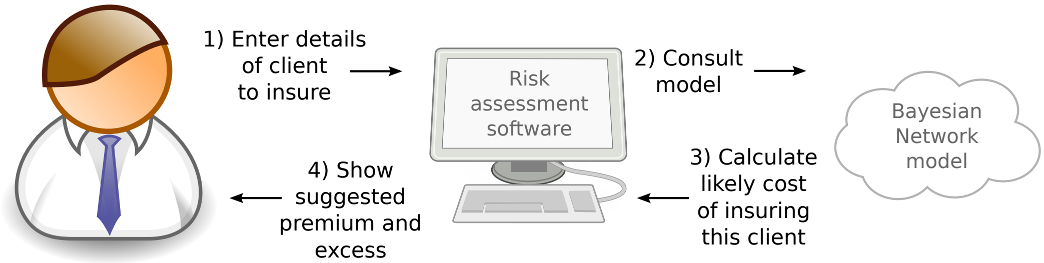

1) Enter details of client to insure
Risk assessment software
2) Consult model
Bayesian Network model
3) Calculate likely cost of insuring this client
4) Show suggested premium and excess

Figure 1.1.: High level overview of an example car insurance risk assessment system using a BN to help make decisions.

This example illustrates important characteristics which are common to most models, even though many different models could have been used for such a system. The actual reasoning process is abstracted from both the end user and the software system. It is not a requirement for either of them to concern themselves with the specific implementation details of the BN model. However, this abstraction allows input to be given, and useful responses to be provided in response.

It is absolutely essential that the underlying model is as accurate and efficient as it can be, although the rest of the system (e.g. user interface, database, etc) is also important. Entire fields of research and industry have been dedicated to the building of models that leverage the ability of computers to aid in decision making. Often the process of producing models requires vast amounts of data, and clever ways to analyse that data. The fields of machine learning and data mining have produced numerous algorithms capable of “learning” models by analysing large data sets (Russell and Norvig, 2010) that have subsequently found favour in artificial intelligence, decision support, and other applied fields.

Research in the areas of machine learning and data mining represents a fast growing body of literature. Existing methods get improved incrementally, several methods may get combined, and occasionally completely new methods arise. The end goal is usually to produce an effective and efficient model, both with respect to the process of constructing the model, and also the ability of the model to be used to perform calculations and hence support decision making. This thesis presents a technique for combining existing techniques, in order to improve the process of eliciting BNs from experts.

1.3.1. Bayesian Networks (BNs)

A particular type of model commonly constructed using either machine learning or KA is a BN. BNs have been successfully applied to solve problems in many diverse fields over the past three decades since being popularised by Pearl (1988). Technically speaking, a BN is "a probabilistic graphical model, which encodes the conditional independence relationships between various random variables" (Friedman et al., 1997, p134). For the purpose of this introduction, the following definition will be used:

> "Bayesian networks are probabilistic models which represent the causal relationships between variables. This in turn allows them to be used to make inferences about the complex interaction between these variables."[1]

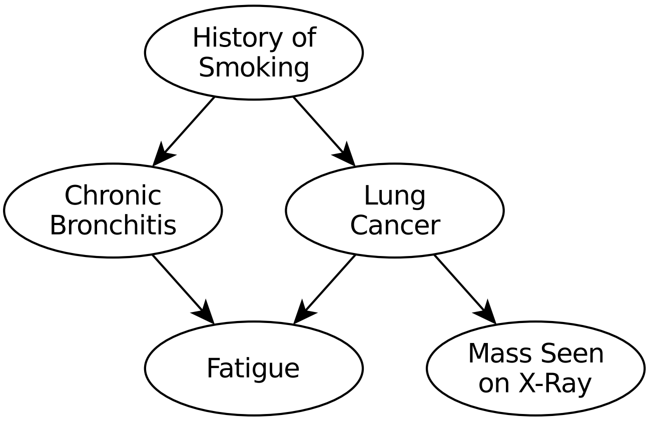


Figure 1.2.: Example "Cancer" Bayesian network (from Cooper, 1999).

Figure 1.2 shows an example BN that models the relationship between *smoking*, *bronchitis*, *lung cancer*, *fatigue* and *x-ray screenings* (Cooper, 1999). This relatively

[1] This definition is intentionally kept brief, and as such does not completely describe BNs. Not all BNs encode causal relationships, nor have every aspect of BNs been described here. For a more detailed definition, see Section 2.2 (p24)

simple model can aid decision makers in several ways. One such way is by providing a better understanding of the problem at hand, with observations such as:

1. A fatigued patient may have either lung cancer or bronchitis. However if a mass is seen on their X-Ray, then the fatigue is likely caused by lung cancer, *not* bronchitis.

As this BN models *causal* relationships, it also facilitates causal reasoning, enabling statements such as:

2. If you have lung cancer, it is more likely that you will also be fatigued

Doctors could propose medical interventions and investigate their effects without resorting to potentially costly, risky, or unethical experimentation:

3. If you were to increase your smoking from one cigarette a day to a pack a day, it will increase the chance of you having bronchitis by $X\%$, and lung cancer by $Y\%$.

As with most computer models, it is common to construct BN models by analysing data sets and identifying the relevant relationships between variables. Many algorithms exist allowing BNs to be created in such a manner (e.g. Buntine, 1996; Friedman et al., 2000; Spiegelhalter and Lauritzen, 1990; Zhang et al., 2012). However, one of the major strengths of BNs is that they do not have to be constructed in a data-driven fashion. This becomes important when the data is not accessible, is not suitable, or there is not enough data (Lucas et al., 2004, p205).

A very popular method for constructing BNs is through KA without the aid of data sets (e.g. Kuikka and Varis, 1997; Przytula and Thompson, 2000; Rumantir, 2003; Falzon, 2006; Chan et al., 2011). These are typically created by interviewing domain experts with the goal of encoding their knowledge into BN models. These models can then be used to reason about particular domains and help solve problems. This approach is often referred to as Knowledge Engineering for Bayesian Networks (KEBN Korb and Nicholson, 2011), which this thesis refers to as traditional KEBN to disambiguate with the new approach proposed.

KA differs from the data-driven approach, because when opting for a data-driven approach to constructing BNs, researchers usually make use of existing commercial off the shelf software for this task such as Netica (Norsys Software Corp, 2016) or BayesServer (Bayes Server Ltd, 2015). When KA is used for BN construction this is not the case. Rather, the number of ways in which elicitation is undertaken is quite diverse, with each knowledge engineer having different preferences for approaching the task of interviewing experts. There are some good textbooks and research articles that provide advice and discuss best practices for KA (e.g. Milton, 2008; Studer et al., 1998) and specifically for the purpose of BN construction (e.g. Kjærulff and Madsen, 2013; Korb and Nicholson, 2011). However, there is less research in developing new approaches using KA to construct BNs, especially compared to the volume of research on data-driven BN methods.

Despite the common usage of KA to elicit BNs, and the textbooks and journal articles providing guidance on how best to conduct such elicitation, there is still little research providing new and improved methods of KA for BNs. Hence, this research introduced a new, prescriptive technique to help BN practitioners and researchers alike construct BNs through expert elicitation.

1.3.2. Knowledge Acquisition

Knowledge Acquisition (KA) is the process of modelling and documenting tacit knowledge from experts, often through interviews, in order to elicit the required knowledge to construct a model (Simon, 1996, p271). The broad goal is to produce a useful model which helps people to make decisions. In this way, KA is similar to Machine Learning (ML) which strives to build models by analysing large data sets for patterns (Russell and Norvig, 2010). The difference is that KA makes use of the education and experience acquired by experts to create the model under construction. Therefore, it stands to reason that KA is often used in situations where large data sets are unavailable or when the expertise of domain experts is deemed to be more useful than the available data sets.

The process of KA can take many forms including semi structured interviews, focus groups, and protocol analysis (Dieste and Juristo, 2011). Some KA tasks may be completed with only a single interview, while others may involve ongoing meetings where new knowledge is elicited each time or previous knowledge is refined (Milton, 2008). Features common to most KA sessions are that one or more analysts direct the elicitation and initiate discussions with domain experts. Additionally, they all share the same goal of eliciting knowledge from experts and encoding it into a model. This ensures that the knowledge persists beyond the employment or availability of a particular expert and is available to aid in training and decision making into the future.

As with most choices, using KA to construct a BN in comparison to data mining has some disadvantages. For example, experts often struggle to convey their knowledge such that it is fully comprehensible to the knowledge engineer, or the knowledge engineer struggles to understand what is being said by the expert (Milton, 2008). Knowledge elicitation sessions with multiple experts have additional issues, such as integrating knowledge from multiple, often conflicting opinions (Clemen and Winkler, 1999). These issues can be overcome by applying formal methods designed specifically to mitigate them. For example, Onwuegbuzie et al. (2009) proposes a method for explicitly documenting the level of consensus in a group and how that level changes over time.

1.3.3. Surveys

This thesis focused on building BNs by administering online surveys rather than analysing data or interviewing experts. As such, it is important to investigate the history of surveys to better understand the benefits and concerns associated with them.

Surveys have historically been used by the social sciences for the purposes of better understanding a large population by sampling only a representative few (Babbie, 1990, p42). This allows inferences to be made about the broader group of people, aiding in decisions made when considering their requirements. Of course, by "population", one may actually mean the literal entire population of a country or the world, but this is usually not the case. It is more common that the population of interest is a subgroup of people such as "university students in Australia" or "people with a family history of cardiac arrest". By carefully selecting and questioning a small proportion of people, it is possible to draw inferences about the broader population under study.

There is a plethora of research on the survey methodology and how to avoid common pitfalls (e.g. Andersen et al., 1979; Babbie, 1990; Baker et al., 2014; Groves and Lyberg, 2010). Bias can be introduced in a number of different ways calling into question the inferences made from survey data. This is such a common theme in the field of surveys that the term "total survey error" has been coined (Andersen et al., 1979; Groves and Lyberg, 2010; Weisberg, 2009) to emphasise that a large part of running a successful survey is mitigating as many of the possible sources of bias (i.e. error) as possible.

Historically, surveys have mainly been used for this concept of extrapolating from a sample to a population:

> "Sample surveys are almost never conducted for purposes of describing the particular sample under study. Rather, they are conducted for purposes of understanding the larger population from which the sample was initially selected." Babbie (1990, p42)

However in recent years, surveys have been finding a new role in research and industry. In addition to their primary use as a sampling tool, they are also used to build models that encode expert knowledge, though infrequently. Some examples are online crowd sourcing "citizen science" projects, which make use of survey responses to increase our knowledge about the world (e.g. Chklovski and Gil, 2005; Raykar et al., 2010). Other examples are more akin to taking traditional KA and considering how it can be augmented by using surveys (e.g. Baker et al., 2014), as was the case with this thesis. This use of surveys is not about understanding the population the sample survey subjects represent, but rather extracting the knowledge that the survey subjects have accumulated in their respective areas.

1.3.4. Summary of Previous Research

This thesis draws on research from the fields of BNs, KA, and the survey methodology in order to propose a new method for constructing BNs. This will in turn enable a wider audience of people to utilise BNs for decision making in their field.

1.4. Research Methodology

The methodology used to undertake this research was informed by the Design Science Research (DSR) method, influenced by research by Walls et al. (1992); March and Smith (1995); Hevner et al. (2004); Peffers et al. (2007); Gregor and Jones (2007). The general principle is that in addition to the natural sciences (e.g. physics, chemistry, etc) which study the world, researchers should also strive to study artificial things produced by humans (artifacts). The study of such artifacts results in both knowledge about the specific artifacts under study, but also knowledge about how best to undertake the future development of other artifacts.

The field of DSR is said to have been pioneered by Herbert Simon with his work "The Sciences of the Artificial" (Simon, 1969, 1st ed.) and later formalised by researchers such as Nunamaker Jr and Chen (1990), Walls et al. (1992), March and Smith (1995), Hevner et al. (2004) and Gregor (2006). The methodology of DSR is still an active area of research and is undergoing improvements in terms of evaluation (Venable et al., 2012) and communication of research results (Gregor and Hevner, 2013).

There were two main artifacts developed during this research project. The first is a new *method* for eliciting knowledge from experts in order to construct BNs using online surveys, termed "Survey Elicitation for Bayesian Networks" (SEBN). Methods are important contributions because they help facilitate the transformation of user needs into system requirements (March and Smith, 1995), allowing the implementation of a working system to satisfy those needs. The SEBN artifact proposed in this research is presented in detail in Chapter 4 and 6. Walls et al. (1992, p49) stress that this type of research should "provide specific guidance to the design process through a prescriptive mode". As such, these chapters present detailed flow charts and accompanying documentation that together prescribe how SEBN can be undertaken. Knowledge engineers wishing to implement SEBN in order to conduct online surveys should be able to do so.

In order to evaluate this method, a second artifact was created. This was an *implementation*[2] of the method in the form of an online survey system, termed "Bayesian

[2] Most DSR papers use the term "instantiation", but this thesis opts for the word "implementa-

Network Elicitator" (BNE). The online survey was administered for the purposes of constructing a BN model to help with car insurance risk assessment. Two evaluation studies enlisted over 100 participants to answer online survey questions and combined the responses into a BN. The goal was to have an existing, published BN (Binder et al., 1997) with which the BN resulting from the evaluation could be compared. The similarities between the evaluation and the existing networks are measured using various metrics described in Chapter 5 and Chapter 7.

Various DSR guidelines are followed throughout the thesis, particularly those from Hevner et al. (2004). Specifically, viable artifacts result from the research (Hevner et al., 2004, Guideline 1), to solve a relevant problem (Hevner et al., 2004, Guideline 2), and they are rigorously evaluated (Hevner et al., 2004, Guideline 3). The research produces clear contributions (Hevner et al., 2004, Guideline 4), and the research is communicated clearly both to researchers interested in building on the method as well as practitioners wishing to implement the technique or indeed make use of the implementation from this research (Hevner et al., 2004, Guideline 7).

1.5. Significance

The significance of this research is that it facilitates BNs to be elicited from experts with the following benefits over traditional interviews:

- Reduced burden for knowledge engineers and researchers alike.
- Encourages a greater number of experts, and hence a greater range of expertise to be incorporated into the resulting BN.

Previous research on KA makes use of interviews for the purpose of eliciting knowledge. This tends to mean face to face individual or group interviews such as workshops or focus groups. To a lesser extent, KA can be conducted via interviews using online video chat, phone or some other technology that allows geographically disperse people to communicate. However whether the interviews are conducted

tion" as it is a common term used in software engineering to talk about realising an abstract concept in a tangible piece of software. This thesis will use the terms interchangeably where required, to fit with existing DSR. This distinction will be discussed in greater detail in Section 3.4.3 (p68) face to face or in an online setting, the process of manually interviewing experts is time consuming. As more people are included in the KA process, it becomes more difficult to manage the project and incorporate the knowledge from multiple experts. By allowing the process of elicitation to be conducted via online surveys, an arbitrarily large number of experts can be included in the KA process without substantially increasing the burden on the knowledge engineer. Although the experts must still be recruited and consulted, there is no longer a need to conduct, transcribe, or analyse interviews with them. This comes at the expense of more in depth or exploratory interviews with each expert.

In addition to reducing the burden on the knowledge engineer interested in constructing a BN, this thesis also proposes methods of reducing the burden on experts. This is achieved in two main ways. Firstly, each expert is only asked a subset of all required survey questions. Secondly, the online survey is constructed in such a way that experts need not be trained in the nuances of BNs in order to contribute their knowledge as is often the case in KA for BNs.

By enabling more experts to participate, it is hoped that this research encourages others to opt for methods of KA that do not restrict the number of experts. Although more opinions can be counter productive in some situations (e.g. when a consensus is required), this does not occur when building a probabilistic model as that uncertainty is able to be encoded in the resulting model. The uncertainty caused by disagreement is encoded in the model itself, to truly represent the uncertainty present in the knowledge of the experts.

1.6. Contributions

This research makes contributions to both theory and practice of KA for BNs. The following paragraphs explain the specific contributions in these areas.

1.6.1. Contributions to Theory

This research contributes to the field of KA for BNs. Although there is limited existing research which makes use of surveys for the construction of BNs, they are not supported with any theoretical discussion of its benefits. For example, Xiaoxuan et al. (2007) propose a technique for using surveys to elicit BNs, however potential pitfalls are not discussed such as the very real potential for their technique to result in cycles in the resulting BN structure, resulting in an invalid BN. In addition, the existing approaches require evaluation to determine if they were indeed a successful alternative to traditional knowledge engineering. To the best of the authors knowledge, this was the first research which thoroughly and rigorously investigated the concept of using surveys for this purpose. This thesis investigated aspects of using surveys to elicit BNs such as resolving cycles in the BN structure, identifying potential portions of the network which can be optimised, and choosing the optimal elicitation technique based on the local structure of the network. It also investigated how to reduce the burden on experts so they are not required to answer an unreasonably large amount of questions, and integrates differing opinions of multiple experts into a single BN. These contributions were made by integrating previous research in the field of the BN construction, KA, the survey methodology, and crowd sourcing. The result is a novel workflow for eliciting BNs using online surveys.

1.6.2. Contributions to Practice

The feasibility of the theoretical contributions was evaluated by building an online system capable of administering surveys which produce BNs. It is released[3] under the GNU GPLv3 license (Free Software Foundation, 2007) to encourage anybody to make use of it and contribute to its further development. The system was used in real life conditions, and managed to satisfactorily administer two surveys consisting of over 100 participants.

1.6.3. Contribution to Methodology

The body of literature concerning the DSR methodology itself is added to at the conclusion of this research. Contributions were made to the DSR method through

[3] BN Elicitator is available from `https://github.com/bn-elicitator/bn-elicitator`

discussions on the topic of “instantiation validity”. This phenomenon arises when the “ad-hoc nature in which artifacts are designed contribute to inconclusive and mediocre findings” (Arazy et al., 2010; Lukyanenko et al., 2014, p322). Section 8.4 (p206) discusses how it applied to this research project, and also proposes additional considerations to help when conducting DSR for other projects.

1.7. Scope of the Research Project

This research projected focussed on constructing BNs. It presented background research on the field of KA, and also made contributions to the field of KA, by using online surveys to perform KA tasks that are traditionally performed using interviews. However this will be limited to KA tasks that are required in order to elicit a BN. For example, the KA task of eliciting probabilities receives quite some attention, however other aspects of KA such as observing experts performing their role is not the focus of this research.

In order to construct a BN, three main types of information need to be elicited in sequence: variables to include in the model, relationships between the variables, and the probability values which are used to parameterise the model. Within the bounds of BN KA, the project focuses on the final two of these. Both the structure and the probability elicitation are discussed in depth, and evaluated using empirical experiments. The ability to elicit variables is not discussed in detail as part of the workflow presented in this thesis, and thus not evaluated. However, it is briefly touched on to show how it fits in with the bigger picture in Section 8.7.4 (p216).

The scope of this research did not allow for multiple refine and re-evaluation stages of the research as is often the case with larger projects. However the concluding chapter comprehensively discusses future research directions based on the lessons learned from the evaluation studies (Section 8.7, p214) . This includes discussion on ideas of how to refine the technique of elicitation, appropriately change the software system, and conduct further evaluation.

1.8. Thesis Structure

The rest of the thesis will be dedicated to motivating, describing, and evaluating a new technique for eliciting knowledge from experts for the purpose of constructing BNs.

Literature Review (Chapter 2, p21) The literature review begins by discussing BNs, how they are used, and how they are typically constructed. KA is then introduced, as is research on the survey method, expert judgement, and combining expertise from multiple experts. It highlights pitfalls and biases known to arise when eliciting knowledge from experts.

Methodology (Chapter 3, p57) This thesis draws inspiration from several DSR theorists. Chapter 3 highlights aspects from Hevner et al. (2004); March and Smith (1995); Nunamaker Jr and Chen (1990); Venable et al. (2012) which are used to frame this thesis. The artifacts which form the contributions of this thesis are then introduced. The chapter also details a set of propositions which this thesis seeks to address. These propositions are used as the basis of the research evaluation which is also discussed in the methodology chapter.

Building BN Structure Through Survey Based Elicitation (Chapter 4, p73) This chapter constitutes (along with Chapter 6) the main contribution of this thesis: a workflow for the Survey Elicitation of BNs (SEBN). This chapter discusses the elicitation of BN *structures*, whereas Chapter 6 looks at eliciting *parameters* of a BN. In order to elicit BN structures using online surveys, this chapter explains what questions are required to be asked during the survey and how responses map to a BN structure. The chapter explains how the number of questions required of experts is minimized in two main ways. These involve constraining the total amount of questions which must be answered in order to build a BN, and only allocating a subset of these questions to each expert. The chapter concludes by discussing how to collate multiple responses to each survey question into a final BN structure. This discussion includes consideration of how to resolve disagreement among experts and how to identify anomalies that arise when multiple responses are collated into a single model.

Evaluating Structure Elicitation (Chapter 5, p103) This chapter discusses an evaluation study which used an implementation of the workflow from Chapter 4 in order to investigate what BNs resulting from SEBN would look like. In order to evaluate the resulting network, it was quantitatively compared to an existing published network. The evaluation also measured how much time was required of each participant while conducting the survey.

Calculating BN Probabilities Through Survey Based Elicitation (Chapter 6, p155) This chapter focuses on eliciting the *conditional probability tables* (CPTs) which are used to parameterise a BN. The chapter adapts specific techniques from van der Gaag et al. (1999), Das (2004), and Saaty (1990) in order to mitigate the problem of combinatorial explosion and the subsequent number of questions required of experts. The allocation of a subset of questions to each expert and a method of collating responses from multiple experts is also discussed.

Evaluating Probability Elicitation (Chapter 7, p175) In this chapter, a second evaluation study is discussed where an online survey was used to elicit the *parameters* of an existing published BN. As with Chapter 5, the resulting BN was quantitatively compared to an existing network and the results documented. Also, the time required of participants was investigated.

Conclusions and Future Work (Chapter 8, p191) Concluding remarks are presented in this chapter including a summary of the thesis and more specific discussion about the contributions of the thesis to both theory and practice. This is accompanied by revisiting the propositions introduced in the methodology, and whether they came to fruition or not. During this research, many avenues of future work were identified, and these are also documented in this chapter.

2. Literature Review

This thesis proposed a new technique for generating Bayesian Networks (BNs) to address some of the shortcomings with existing approaches. In order to understand what these shortcomings are and how they will be addressed, this literature review discusses BNs and how they are typically constructed using either data analysis or expert elicitation. As this thesis deals predominantly with the expert elicitation approach, this review continues on to investigate the field of Knowledge Acquisition (KA). In particular, it is interested with the issues that should be considered when combining expertise from multiple experts from the field of KA, and multiple lay people from the field of crowd sourcing. In order to address some of the concerns with current KA approaches to eliciting BNs, this thesis adopts the online survey methodology in preference to face to face interviews. Thus, this chapter concludes with a review of the survey methodology and how it relates to the field of KA.

2.1. Computer Models for Problem Solving

A BN is a type of computer model which can be employed to solve problems. To better understand why one would choose a BN to solve a particular problem, this section first discusses computer models in general.

The Oxford English Dictionary defines a model as:

> “A simplified or idealized description or conception of a particular system, situation, or process, often in mathematical terms, that is put forward as a basis for theoretical or empirical understanding, or for calculations, predictions, etc.” Oxford (2016)

To clarify this somewhat vague definition, it is helpful to look at a common example of a model, e.g. a model train. A model train hides the complexity of a real, full size, working locomotive in order to show how a train works fundamentally. Thus, it is *a simplified or idealized* implementation *of a particular system* (a train) which allows for a better *empirical understanding* of how a train works. The concepts which are simplified or abstracted away include that of the internal combustion or steam engine that drives the wheels, the braking mechanisms, and various other ideas. The abstraction is still very useful though, because it helps children to understand how a train works in the real world, how the locomotive travels on two rails in order to move, and how several carriages can be chained together to make a longer train.

The same applies to *computer* models, i.e. algorithms or software which provide an abstraction over a set of complex concepts, to help reason about and understand them. Consider the example of speech recognition in humans, for which computer models have been used to help understand. Despite physiologists and psychologists not having a comprehensive understanding of the complex machinery in a human brain which performs speech recognition, computer scientists in the field of artificial intelligence (AI) have not been deterred. One particularly successful computer model (and one of the earliest applied to AI) used to help computers recognise speech is an artificial neural network (ANN, first proposed by Mc Culloch and Pitts, 1943). ANNs provide useful albeit extremely simplified abstraction of the way a real brain works. As such, these *simplified or idealized* ANN computer models have had good success in helping people conduct speech recognition using a computer.

2.1.1. Constructing Computer Models

There are two main ways in which models are constructed, through analysing large historical data sets, or through knowledge acquisition[1]. In addition, there is

[1] Constructing models is one thing, but maintaining them into the future, adapting to new and changing situations, or refining them when new information comes to hand is equally important. This section will deal specifically with construction of models, but many of the same problems and solutions are relevant to maintaining the models.

a third option which is to combine analysis of historical data sets with KA. The choice of which method is used is primarily influenced by the type of model which is required in a given situation. As a secondary consideration, some models can be constructed by either approach, so it becomes a decision which needs to be made by the model builder(s) on a case-by-case basis. Primarily though, the choice of how to build a model is constrained by the type of data available.

Constructing Models Using Historical Data

Techniques for constructing models by analysing historical data fall under the field of Machine Learning.

> *Machine Learning* is the field of scientific study that concentrates on induction algorithms and on other algorithms that can be said to “learn.” (Kohavi and Provost, 1998)

Machine Learning (ML) deals with algorithms or software designed to identify patterns in a set of data, with the purpose of being able to make useful predictions about unseen data. As with most things, there are many and varied definitions, but this will suffice for the purpose of this research[2]. Once patterns have been identified in a data set, they can often be encoded into a computer model to help make sense of new, previously unknown data.

Constructing Models Using Knowledge Acquisition

Knowledge Acquisition (KA) is the process of a human expert imparting their knowledge and experience into a model. This is useful in situations where historical data is unavailable, inappropriate, or insufficient. It is also appropriate when the model building process is more exploratory, and the decision as to what is required may not be known until consultation with experts is conducted. Section 2.3 will provide a much more comprehensive view of KA in general, and Section 2.2.3

[2] Note that while this literature review refers to Machine Learning, there is a sister discipline termed Data Mining, which focuses on similar goals. The task of Data Mining can be seen as the broader process of knowledge discovery of which ML is one component (Kohavi and Provost, 1998).

discusses KA specifically in relation to the construction of BNs, which is the focus of this research.

Compared to ML, KA research tends to focus more on general approaches to eliciting knowledge from experts. There are some publications on the specifics of KA for constructing specific types of models (e.g. Kjærulff and Madsen, 2013, for BNs). However these are amongst the minority as compared to those that only provide direction for conducting KA in general, without specifying the type of model to construct (e.g. Milton, 2008).

The research in this thesis will provide a KA workflow specific to BNs. Specialising in this way means that the workflow can contain very specific optimisations for the BN use case, without making compromises in order to stay generic.

2.2. Bayesian Networks (BNs)

This section focuses specifically on one model - the Bayesian Network (BN), which is the type of model that this thesis contributes to. The following sections will introduce what a BN is, why one would choose to use a BN, and when one may choose *not* to make use of them. These are then followed by a review of the ways in which BN models are constructed. This is used as a motivation to propose a new technique for creating BNs presented in this research.

2.2.1. What is a BN?

BNs are a type of computer model that facilitate probabilistic reasoning. They have a graphical component, which can be seen in Figure 2.1, and each node in the graph has a table of probabilities associated with it known as a Conditional Probability Table (CPT), which is not represented graphically. A BN is a way to represent a *factorised joint probability distribution*. To illustrate this concept, consider the (non-factorised) join probability distribution of four variables of interest in a simplified model of smoking and cancer. The model encodes the probability that a patient is subjected to 4 different conditions: that they *smoke*, have

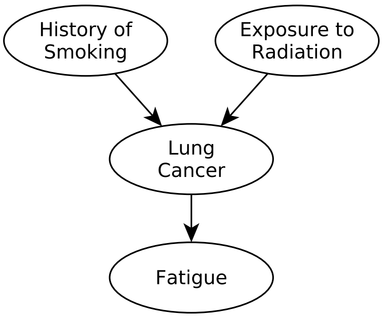


Figure 2.1.: Example BN modelling the relationship between smoking, radiation, cancer and fatigue.

been *exposed to radiation*, have *cancer* and suffer *fatigue*. The joint probability distribution which this represents is:

$$\Pr(Smoking, Radiation, Cancer, Fatigue)$$

The first three of these variables have two states (true or false) and the fourth variable, fatigue, has three (low, medium, or high). This results in a total of 24 (2×2×2×3) possible combination of scenarios a patient could be in. Some are more likely than others, and as such, the probability of each 24 combinations needs to be calculated for this distribution to be useful in a model. As the number of variables increase, the total number of probabilities required to parameterise the model increases exponentially. Thus, it is usually desirable to factorise this distribution to make it more manageable. Factorising joint probability distributions is similar to factorising other quantities in mathematics. It is the process of decomposing the joint probability into a series of conditional probabilities, than when multiplied together, result in the original joint distribution.

Using the chain rule of probability, this joint probability distribution can be transformed into the following conditional probabilities:

$$\begin{aligned}&\Pr\left(Smoking, Radiation, Cancer, Fatigue\right) = \\ &\Pr(Smoking)\times\Pr(Radiation|Smoking)\times\Pr(Cancer|Radiation, Smoking)\times \\ &\qquad\qquad\Pr(Fatigue|Cancer, Radiation, Smoking) \qquad (2.1)\end{aligned}$$

The total number of probability values required to represent this distribution is now $2+(2\times2)+(2\times2\times2)+(3\times2\times2\times2) = 38$. This is in fact larger than the 24 probabilities required for the join distribution, but it becomes easier to factorise in this form. To illustrate, consider the *fatigue* variable. If there is no information about the patient, then the model will make a best guess estimate about how likely it is that they are fatigued. However, if someone was to say that they smoke, then the belief that they suffer fatigue would increase - as smoking often causes cancer and fatigue is a common symptom of cancer. If instead of informing about their smoking habits, they were instead known to have cancer, the belief in them being fatigued would increase for the same reason.

However, if it was *already known* that the patient had cancer, and it subsequently became known that they *also* smoke, it would *not* change the belief in that person being fatigued. This is because the only mechanism with which smoking habits informed the belief that the patient was fatigued is because it increased their chance of cancer. If it was already known that they have cancer, the knowledge about smoking *doesn't increase or decrease the belief that the patient is fatigued*[3].

This example illustrates the *conditional independence* property between the causal variables in a BN. *Fatigue* and *smoking* are independent to one another conditioned to the fact that the patient has *cancer*. Additional information about the smoking habit of the patient will not change the belief of the patient being fatigued:

[3] This may not be true, it may be that smoking in itself directly causes fatigue. However, as models are simplifications of complex phenomena, the decision has been made to only model the effect of smoking on cancer, not its effect on fatigue. This is an example of a subjective decision that is often made when building BNs or other models, and which requires careful consideration. However, as this is an example to illustrate factorising probability distributions, this consideration will be ignored for now.

$$\Pr(Fatigue|Smoking, Cancer) = \Pr(Fatigue|Cancer)$$

If the patient *doesn't* has cancer though, then the two variable are dependent to one another, i.e. the smoking habit of the patient will help to determine the change of the patient being fatigued.

More generally, variable A and C are "conditionally independent given B" if $\Pr(A|B, C) = \Pr(A|B)$. Note how the distribution of $\Pr(A|B)$ has less parameters than $\Pr(A|B, C)$, which makes it easier to model. Therefore, if the cases of conditional independence were identified and factorised out of the distribution of Eq. 2.1, the whole model will become simpler and easier to manage. The end result is that Eq. 2.1 can be factorised into the more manageable:

$$\begin{aligned}&\Pr(Smoking, Radiation, Cancer, Fatigue) \approx \\ &\quad \Pr(Smoking) \times \Pr(Radiation) \times \Pr(Cancer|Radiation, Smoking)\times \\ &\qquad\qquad Pr(Fatigue|Cancer)\end{aligned} \tag{2.2}$$

Note the similarity between Eq. 2.2 and the network in Figure 2.1. Each conditional probability implies a directed arc in the graph, so $\Pr(Fatigue|Cancer)$ means there is a directed arc from $Cancer \rightarrow Fatigue$ in the BN.

In the basic example shown here, there was only four variables, yet the number of parameters the model needed was reduced by 25% from 24 to 18. This reduction quickly improves as more variables and arcs are added, because the number of probability values required to represent a joint probability distribution increases exponentially with the number of variables. As such, BNs have become a popular way to model such distributions.

2.2.2. Considerations When Choosing a BN in Preference to Other Models

Nicholson et al. (2008, p35) discuss several features of BNs that make them a good choice of model for many purposes:

> “BNs provide a clear graphical structure with a natural causal interpretation which is intuitive”

This fact was leveraged by van der Gaag and Helsper (2002) in eliciting a BN, as their experts did not have formal training in probability theory. Causal interpretations are important to encourage users to trust the model when using it. If a model is going to be used to recommend some course of action that requires resources to be committed, then it is helpful to know the justification for it (Buchanan et al., 2006, p97). Additionally, when enlisting experts to help construct the models (Section 2.2.3) it would be better if they could visualise and comprehend the model they are producing. Experts enlisted to construct a model, and those who will use the resulting model for problem solving, require trust in that model for which intuitive interpretations help.

> “BNs provide good estimates even when some predictors are missing”

BNs are able to adapt to situations with missing information, given their probabilistic nature. This is important, as missing data is a significant problem, and most data analysis processes were not designed with missing data in mind (Schafer and Graham, 2002).

> “Separation of prior distributions from other parameters, allowing adaptation to new populations”

When new data sets become available, it is possible to re-learn the conditional probabilities which form the parameters of an existing network, as long as the structure of the network (the dependencies between variables) is retained.

> “BNs can incorporate additional data including subjective expert knowledge”

There are many computer models that can be built by analysing historical data, but only a small number can be built in the absence of these data sets.

Of course, with any decision about which model to make use of to solve a given problem, there should be some contra-indications that help guide when *not* to use a particular solution. Despite BNs versatility and the fact they have successfully been applied in many areas, this does not mean that they are always the best tool for the job.

Some researchers have disputed the conclusions discussed above. The fact that “BNs provide good estimates even when some predictors are missing” is challenged by the results of building the HEPAR II network (Oniško, 2008), where performance was found to decrease linearly with the amount of noise in the data used to train the networks.

Adapting the structure of a BN in light of new data is something for which there is not sufficient research into (Jensen and Nielsen, 2007, p214). This is refuted by Korb and Nicholson (2011, p357) who suggest that using the CaMML approach (Wallace and Korb, 1999) would enable adaptation of BN structures. Despite this refutation, it still appears that the adaptation of structure is not able to be dealt with in a manner similar to that of probabilities, whereby new information gradually changes the model. Rather, any change in structure would require a new set of parameters, which could be completely different from the previous parameters. Thus any change to the structure may run the risk of making redundant the work which was previously done to elicit or calculate the probabilities.

2.2.3. Typical Ways in Which BNs are Constructed

There are three main steps in building any BN

1. Identify the variables to be included in the network.
2. Decide on the structure of the network.
3. Parametrise the CPTs of each node.

They are usually done in this order, because the later cannot be completed without the former first being completed. It is very important that the network structure is completed before the CPTs. For example, consider a network structure which is incorrect. Considerable time and effort may be spent calculating the CPTs for each node, only to find that when the structure is reorganised, this information becomes outdated and irrelevant, and new CPTs are required. Thus it is important to get the structure of the network correct first.

As was discussed briefly in Section 2.1.1, there are two main ways in which computer models are built: algorithmically by using ML to analyse historical data sets, or manually using expert elicitation. Construction of BNs is no different, and for each of the three main steps of creating a BN described above, either historical data or expert elicitation approaches can be used. The following section will discuss approaches to creating BNs that utilise historical data sets. However, unlike some other predictive tools constructed by analysing historical data, the absence of suitable data does not preclude a BN from being used to solve a problem. The next section will discuss how expert knowledge elicitation can also be used to produce a BN. Finally, there is no shortage of approaches which combine both data and experts in a hybrid method to construct BNs and these will be mentioned below.

Constructing BNs Using Historical Data

Historical data about a particular problem domain can be used to learn the relationships between variables, based on statistical properties of the data set. The two general methods are search and score based, and constraint based (Kjærulff and Madsen, 2013). Search and score based methods produce several complete BNs, which are then assigned a score ranking how closely they represent the distribution of the available data (e.g. Chickering, 2002). Constraint based approaches start with individual variables and incrementally build an entire BN by adding new relationships as they are identified in the data. They apply "knowledge of conditional independencies to make inferences about what causal relationships are possible" (Korb and Nicholson, 2011, p184). Examples of these inductive learning algorithms include the search and score based K2 algorithm (Cooper and Herskovits, 1991) and the constraint based PC algorithm (Spirtes et al., 2000).

Some requirements for the use of historical data to produce BNs is that it exists, is accessible, is suitable, and that there is enough (Lucas et al., 2004, p205). Non-existent data is a common issue when a BN is being constructed for a once off event or process, such as to help make environmental policy decisions (e.g. Chan et al., 2010), or when the required data has never been collected (e.g. Pellikka et al., 2005). Inaccessible data arises when there are privacy or other concerns which prevent data being used, such as with medical records. Unsuitable data arises when there is indeed data that has been collected, but it cannot easily be processed analytically, such as in-depth interviews or multimedia data, or it does not include the variables for which the BN aims to encode (Lucas et al., 2004, p205). Finally, many techniques require large amounts of data in order to confidently identify the parameters of a network, and the available data sets may not be large enough (Lucas et al., 2004, p205). The following section introduces expert knowledge elicitation as an alternative tool for constructing BNs when any of the above conditions arise.

Constructing BNs Using Expert Elicitation

When no suitable data is available, or when the knowledge engineer deems it appropriate, then knowledge elicitation can be used instead of historical data to construct BNs. Traditional "Knowledge Engineering for Bayesian Networks" (KEBN) usually involves interviewing experts to transfer their knowledge and experience into a BN. There are many good textbooks on the topic of KEBN, including Kjærulff and Madsen (2013) and Korb and Nicholson (2011).

The three steps described earlier, identifying variables, defining structure, and parameterising CPTs are also performed in KEBN. As with the data driven approach, they should be conducted in sequence, in order to prevent changes to the structure invalidating previously calculated CPTs. The process is based on that of KA, which has been used successfully for decades to construct expert systems and other models (e.g. Shortliffe et al., 1984), and which will be discussed further in Section 2.3.

KEBN is usually closer to constraint rather than search and score based methods discussed above, with a single BN being constructed after experts identify the relevant relationships. There are exceptions to this rule, for example Pellikka et al. (2005) effectively produced eight different networks which were then combined together based on scores assigned to each arc. The difference from the approach using historical data is that the decision to include a relationship in the model is based on a persons expertise, rather than conditional independence relationships identified in the data.

KEBN involves the knowledge engineers interviewing each of the experts. This can be either in one-on-one interviews (e.g. Kuikka and Varis, 1997; Pollino et al., 2007), or in workshops with multiple experts (e.g. Chan et al., 2010), and is more often than not iterative. If one-on-one interviews are used, a large number of interviews are required, and the results from each one must be integrated by the knowledge engineers. If workshops or focus groups are conducted, then the knowledge engineers need to organise a single time when all experts can come together to discuss the BN. It also involves dealing with potential sources of bias discussed further in Section 2.3.4.

Constructing BNs Using a Hybrid Between Historical Data and Expert Elicitation

The BN literature also includes methodologies advocating mixing algorithmic and KEBN approaches. Historical data is good for when there is a lot of data available for which to identify patterns within, while KEBN is good when there is a wealth of expertise which can be elicited from experts in a field. It is only natural that hybrid approaches which blend the best of both have emerged. However, it has been noted that blindly combining multiple sources of information may not be the best method, and principled methods should be developed to deal with such combinations of data (Druzdzel and Díez, 2003).

There are a few different ways to approach producing a hybrid method. Although it is conceivable to use an algorithm to initially produce a network structure which is then refined by experts, almost every approach focuses on the more practical alternative of asking experts to produce some prior information or network structure, then refine that by analysing historical data (e.g. Flores et al., 2011; Gambelli and Bruschi, 2010; Heckerman et al., 1995).

An approach which asks an expert to produce the structure, and then making use of data for calculating the CPTs is put forward by Druzdzel and Díez (2003). The hybrid approach put forward by Gambelli and Bruschi (2010) asks the expert to place theoretical constraints on possible relationships. From here, an algorithmic approach is used to elicit a structure which adheres to those constraints. A more Bayesian approach is proposed by Heckerman et al. (1995), whereby an expert specifies the structure of a BN, and that structure and a data set are used as prior knowledge to "correct" the proposed BN. The final BN in this case tends to resemble the network the expert proposed, and should therefore have meaningful causal relationships defined by an expert, as well as interesting relationships identified by the algorithm which the expert may not have known about.

2.2.4. Eliciting the Structure of a BN From Experts

Once all of the variables of interest have been identified (a topic which is beyond the scope of this thesis), then they must be combined into a BN structure. The way in which experts are interviewed in order to elicit this structure varies greatly, as evidenced by the plentiful research into using BNs to solve specific problems (e.g. Martin et al., 2005; Pollino et al., 2007). This section will discuss some common techniques for eliciting the structure of a BN.

Causal Relationships Although arcs in a BN technically represent conditional (in)dependence, many practitioners treat them as something more akin to causal relationships (Heckerman, 1997; Spirtes et al., 2000). Figure 2.2 shows a fragment of the BN from Binder et al. (1997) which has the causal relationships: $Vehicle\,Age \rightarrow Value \rightarrow Theft$. As with many causal relationships, it is also true that these causal relationships also encode the correct conditional independence relationship between the three variables. That is, $Theft$ is conditionally independent of $Vehicle\,Age$ if the $Value$ of the vehicle is known. Thus, it is preferable to consider the option of causal arcs as they are the simplest type of relationship to reason about (Korb and Nicholson, 2011; Kjærulff and Madsen, 2013).

Parent Divorcing One of the common modelling mistakes described by Korb and Nicholson (2011, p319) is to specify too many parents for a given node. This causes

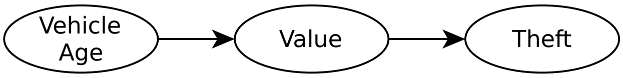
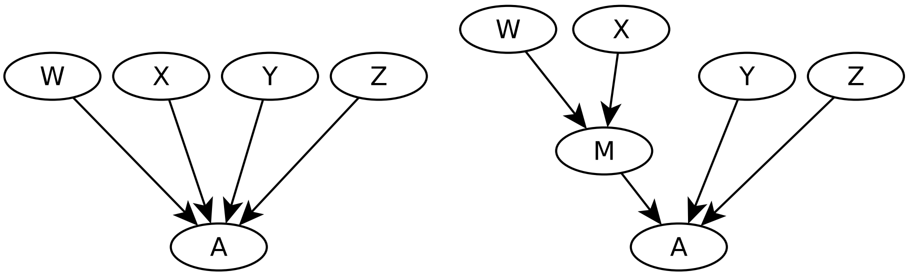


Figure 2.2.: Example of causal arcs which also represent the correct conditional independence relationship between the three variables.



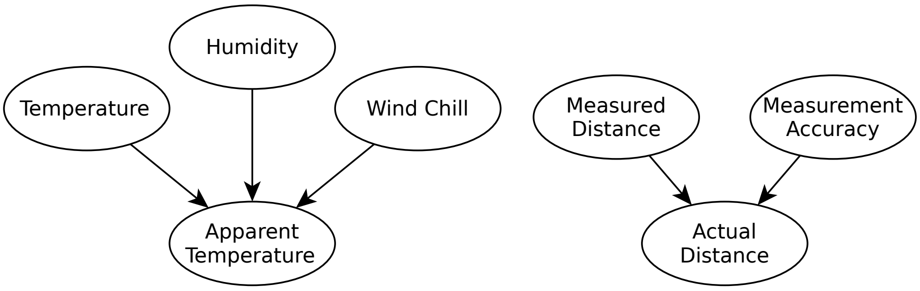
(a) Many parents, causing a potentially large CPT for the child node A.

(b) Although another variable and arc is introduced, the overall complexity of the BN is reduced.

Figure 2.3.: Example of parent divorcing.

the size of the CPT for that node to increase dramatically, making it harder to elicit accurate probabilities. Olesen et al. (1989) proposed a solution called *parent divorcing*, advocated by both Korb and Nicholson (2011, p319) and Kjærulff and Madsen (2013, p192). This is shown by "introducing an intermediate node that summarizes the effect of a subset of parents on a child" (Korb and Nicholson, 2011, p319). An example of parent divorcing is shown in Figure 2.3. Even though it results in an extra node and an extra arc in the network, it substantially reduces the number of parameters required for the child nodes CPT.

BN Idioms In the same way that Gamma et al. (1994) famously proposed software developers make use of design patterns to help solve recurring problems, Neil et al. (2000) propose the use of *idioms* as "a library of patterns for the BN development process" (Neil et al., 2000, p14). Instead of asking about specific conditional independence relationships between variables, knowledge engineers can try to identify a suitable high level idiom to describe the relationships between certain variables. For example, the *Definitional/Synthesis Idiom* dictates that multiple parent variables together define the state of a child variable. For example, in Fig-



(a) Definitional/Synthesis Idiom **(b)** Measurement Idiom

Figure 2.4.: Examples of two of the five idioms proposed by Neil et al. (2000).

ure 2.4a the *Apparent Temperature* node is wholly defined by the *Temperature*, *Humidity*, and the *Wind Chill*. If a node depended on all three parents, then introducing the *Apparent Temperature* node would be a good way to divorce the parents. Another example is the *Measurement Idiom* whereby a variable representing an imperfect measurement is explicitly modelled as uncertain in the BN by introducing a variable to model the accuracy of the measurement instrument (Figure 2.4b). Both of these examples, as with the other idioms proposed by Neil et al. (2000) tend to arise in many situations when modelling BNs.

Specifying BN Arcs via an Adjacency Matrix Xiao-xuan et al. (2007) and Flores et al. (2011) both present methods which asked experts to fill in an $n \times n$ adjacency matrix where n is the number of variables and each cell denotes one of the possible *Parent* → *Child* relationships in the resulting BN. Xiao-xuan et al. (2007) directly combined the results of multiple experts, and where the majority of experts agreed a network existed, an arc was added to the BN. The approach by Flores et al. (2011) is a hybrid learning algorithm where entries in the matrix correspond to priors feed into an algorithm for inducing the BN structure. These approaches require experts to consider n^2 cells in the matrix which can quickly become unmanageable.

Variable Classes Kjærulff and Madsen (2013) present a technique for constraining the possible relationships that need to be investigated by a knowledge engineer and experts during elicitation. This is done by assigning variables to one of four classes: Background, Mediating, Problem, or Symptom. These are shown in Figure 2.5 including the dependencies they exhibit. If one is to assume that the arcs in a network are causal, then there is no need to model a symptom variable having a causal influence on a problem variable in the same BN. Although it is possible to conceive of situations when this may arise, it is worthwhile applying this simplifying modelling assumption in order to reduce the number of arcs that could possibly be added to a BN.

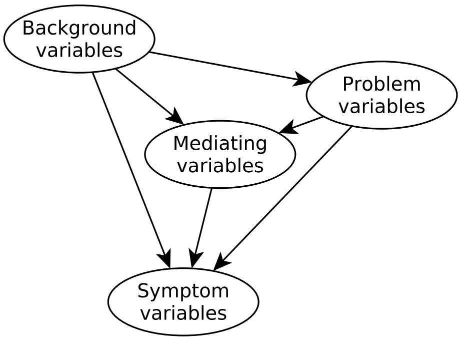
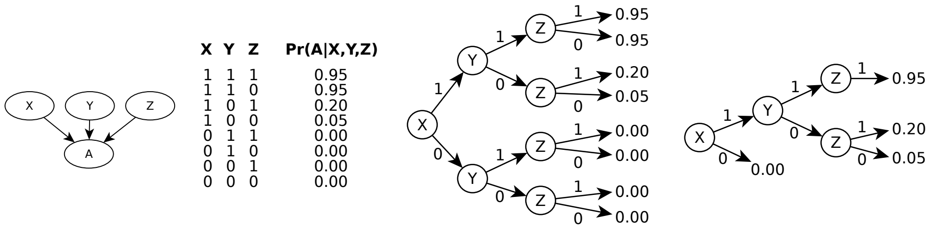


Figure 2.5.: Four general classes of variable, and the logical dependencies between them proposed by Kjærulff and Madsen (2013, p152-154).

2.2.5. Eliciting the Probabilities Required for a BN

CPT elicitation is generally considered the most time consuming part of BN elicitation (Druzdzel and van der Gaag, 2000). This is due to the size of a CPT increasing exponentially with the number of parents to condition on. As a result of this combinatorial explosion, when eliciting large CPTs using historical data, large amounts of data is required. When eliciting CPTs from experts, large amounts of time is required. Buntine (1991) describes the problem as follows:

> “While full conditional joint distributions are more general than any other model, their specification requires an exponential number of parameters. When estimating parameter values from data, this can be a severe problem as it is when trying to elicit the same probabilities from an expert. One way around this is to introduce approximate distributions of lower dimension” (Buntine, 1991, p58)



X	Y	Z	Pr(A|X,Y,Z)
1	1	1	0.95
1	1	0	0.95
1	0	1	0.20
1	0	0	0.05
0	1	1	0.00
0	1	0	0.00
0	0	1	0.00
0	0	0	0.00

Figure 2.6.: Example of a factorised PET requiring 50% of the probabilities than a CPT (adapted from Friedman and Goldszmidt, 1999, p4).

In addition to combinatorial explosion, Jenkinson (2005) discusses the additional problem of bias and errors that arise when asking human experts to judge probabilities. This section discusses specific techniques for eliciting CPTs by approximating distributions of lower dimensions and leaves the discussion of bias and errors due to human judgement to Section 2.3.4 (p46).

Probability Estimation Trees (PETs) By representing a CPT as a probability tree instead of a table, it can more easily be factorised, reducing the number of parameters required to represent it. Taking advantage of the so called "local structure" of a BN node (compared to the global structure of an entire BN) can result in a reduction in the number of probabilities requiring elicitation. Shown in Figure 2.6 is an example of a factorised PET requiring 50% of the parameters of its non-factorised counterpart. PETs have been used to simplify CPTs when reasoning with already existing BNs (Martínez et al., 2002), as well as to improve the probabilities learnt from data (Friedman and Goldszmidt, 1999). However applying PETs to the task of eliciting CPTs from experts has received less research attention.

Analytic Hierarchy Process Yager (1979); Monti and Carenini (2000); Chin et al. (2009); Hughes (2009) all applied the Analytic Hierarchy Process (AHP, Saaty, 1977) to the task of probability elicitation. AHP is used to perform pairwise comparisons between two variables. Only two pieces of information for each pair are required:

1. Which variable is more important

2. The ratio of how much one variable is preferred over the other, as a number between 1 and 9

With this information, Yager (1979) proposed a method of eliciting probability values of the form $\Pr(A)$ from experts. Instead of an expert having to compare and reason about all states of A at once, especially when n is large, they need only compare two probabilities at a time to elicit which is more likely and how much more likely. Combining each of these pairwise comparisons results in a set of probabilities which should be coherent and require less cognitive effort from the experts.

More recent work by Chin et al. (2009) also made use of pairwise comparisons to elicit probabilities, this time in the context of BNs. Their approach is similar to that of Yager (1979), with the difference that they are interested in conditional probabilities of the form $\Pr(A|B, C)$ rather than the simpler $\Pr(A)$. They based their work on the simplifying assumption introduced by Kim and Pearl (1983) that $\Pr(A|B, C) \approx \alpha \times \Pr(A|B) \times \Pr(A|C)$ where α is a normalizing constant. However, Kim and Pearl (1983) explain that this assumption only holds in certain situations. Given that Chin et al. (2009) did not take this limitation into account, more work is required to determine if their approach is suitable for eliciting CPTs.

Weighted Sum Algorithm

Das (2004) proposed the weighed sum algorithm as another approach to ease the burden of eliciting CPTs from experts. It relies on excluding conditional probabilities that seem unnatural and instead interpolating them from more meaningful probabilities. This is done by first asking experts to specify the set of parent states which make the most sense. These are termed Compatible Parent Configurations (CPCs). Probabilities for each of the child node states are then elicited conditioned on each of these CPCs, rather than on every possible combination of parent states. In addition, the relative weights of each parent are elicited indicating which parents have the greatest influence on the child. From this information, the remaining conditional probabilities are calculated as:

$$\Pr(A = a|X = x, Y = y, Z = z) \approx w_x \times \Pr(A = a|CPC(X = x)) + \\ w_y \times \Pr(A = a|CPC(Y = y)) + w_z \times \Pr(A = a|CPC(Z = z))$$

where w_i is the weight of parent i relative to the other parents, in terms of influence on the child variable. To illustrate, take the car insurance network (Binder et al., 1997). The *CarValue* variable is influenced by *Mileage*, *VehicleAge* and *CarType*. If the entire CPT was to be elicited directly, then the following question would eventually need to be answered:

> "What is the probability of the cars value being between \$10k and \$20k
> if it has been driven over 100,000km and it is a new hatchback?"

Otherwise known as $\Pr(CarValue = 10k - 20k|Mileage > 100k, VehicleAge = new, CarType = hatchback)$. However it is unlikely that a new car would have been driven 100,000km. As such, asking this question is likely to cause some confusion as the experts are being asked to assess a situation that they likely have never confronted, nor could they reasonably make sensible guesses at a probability value for this event. Applying the weighted sum algorithm, only the set of parent states which make sense to the expert would be elicited. For this example, the values requiring elicitation are:

- $CPC(Mileage > 100k)$
- $CPC(VehicleAge = new)$
- $CPC(CarType = hatchback)$
- $w_{Mileage}$
- $w_{VehicleAge}$
- $w_{CarType}$
- $\Pr(CarValue = 10 - 20k|CPC(Mileage > 100k))$
- $\Pr(CarValue = 10 - 20k|CPC(VehicleAge > new))$
- $\Pr(CarValue = 10 - 20k|CPC(CarType = hatchback))$

The assumption behind the weighed sum is that each of these values in itself represents a coherent question that an expert can reason about. For the CPC elicitations, they need to explicitly choose situations that seem reasonable, such as the most likely *VehicleAge* and *CarType* if the *Mileage* > 100k. The relative weights are calculated by asking which of the three parents influence *CarValue* the most. The end result is three probability elicitations, each conditioned on a different CPC. Thus, they are only ever conditioned on sets of variables which the expert believes to be sensible.

The weighted sum algorithm results in a reduction (except for very small CPTs - e.g. 4 values) from $s_c \times \prod_{i=1}^{n} s_{pi}$ to $\sum_{i=g1}^{n}(s_{pi} + s_{pi} \times s_c + 1) - l$, where s_c is the number of states the child node can take, s_{pi} is the number of states parent i can take, n is the number of parents, and l is the number of parent configurations which are the same.

Fragment of Text and Probability Scale To address the slow rate at which experts are able to elicit large numbers of probabilities, van der Gaag et al. (1999) proposed a scale to use in place of eliciting specific probabilities. This technique has also been advocated by Korb and Nicholson (2011). Figure 2.7 shows the seven different options on the scale which are shown to experts, and a human readable label explaining what each item on the scale represents. van der Gaag et al. (1999) found that this increased the rate at which experts could elicit probabilities, and experts were purported to find it simpler than other methods of probability elicitation they had used before.

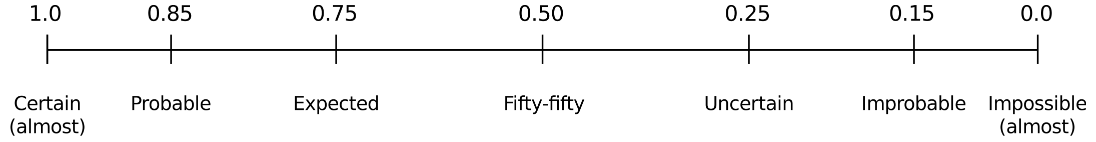


Figure 2.7.: “The Fragment of Text and Probability Scale for the Assessment of the Conditional Probability” adapted from van der Gaag et al. (1999).

2.2.6. Summary of BNs

BNs have seen continued use as problem solving tools in a variety of disciplines. This section described in detail what a BN is and how they are used. They can be created by analysing historical data or by expert elicitation, with this thesis proposing a new expert elicitation approach. The literature is rife with several techniques for eliciting the structure and the probabilities of a BN from experts. One thing that all these techniques have in common is that they were conceived to address some sort of problem that arises when eliciting knowledge from experts. Another thing they have in common is that they require a knowledge engineer, and usually face to face interviews with experts. The following section will investigate how these face to face interviews usually take place, and issues that tend to arise when conducting such elicitation sessions.

2.3. Knowledge Acquisition

The concept of expert elicitation discussed in the previous section is essentially synonymous with the field of Knowledge Acquisition (KA). A useful definition of KA is:

> "The task of giving an expert system its knowledge (i.e., eliciting and codifying it)" (Buchanan et al., 2006, p97)

Although this is able to be done via machine learning, the term KA predominantly refers to a task that is conducted by knowledge engineers[4].

KA was traditionally used for constructing expert systems (ES), which are "computer programs that exhibit some of the characteristics of expertise in human problem solving, most notably high levels of performance." (Buchanan et al., 2006, p87). They represent a quite explicit and tangible representation of an experts knowledge in the form of a computer model. As such, the early attempts

[4] Note that this research uses the terms KA and *expert elicitation* interchangeably. Another term synonymous with KA in the literature is *knowledge elicitation*. This research prefers *acquisition*, in order to prevent confusion when using the acronym KE to refer to "Knowledge *Engineering*" (which includes KA as a sub-task).

to acquire knowledge from experts was seen as very much a transfer of knowledge from the expert to a computer model (Studer et al., 1998). However, as expert systems became more popular, more complex, and harder to construct, the task of KA shifted to modelling the thought process of the expert (Studer et al., 1998).

After proving worthwhile in the construction of expert systems, KA has subsequently found favour in the construction of many other type of computer models. For example, BNs often use expert elicitation in order to be constructed (e.g. Laskey and Mahoney (1997); Henrion (1987); Mahoney and Laskey (1996); Xiao-xuan et al. (2007)), as do ontologies (e.g. Fernández-López et al., 1997), as well as statistical, mechanistic, and other forms of probabilistic models (Krueger et al., 2012).

2.3.1. Relationship Between KA and Knowledge Engineering

KA is one task in the broader field of Knowledge Engineering (KE), a term which defines the entire process of building a computer based solution to a problem. KE encompasses the whole process from planning, requirements gathering, model building, implementation of a computer system, training, and maintenance (Eriksson, 1992). KA is usually concerned with the *requirements gathering* and *model building* phases of KE. However, there is not always a distinct separation between KA and KE, given KA forms such an integral part of KE (Eriksson, 1992, p99).

The process of KE is not dissimilar to the field of software engineering (Eriksson, 1992), where careful thought is put into how a system will be built and maintained, before work is begun on building it. Indeed, Mahoney and Laskey (1996) and others make heavy usage of the relationship to software engineering in order to describe how KE can be used to construct BNs.

2.3.2. Techniques used in KA

Hoffman et al. (1995) and Dieste and Juristo (2011) each provide comprehensive taxonomies of various techniques used in KA, organised into a hierarchical manner.

Hoffman et al. (1995) describes tens of different KA techniques, each of which is categorised into one of three broad categories:

1. Analysis of the tasks that experts perform: "What do experts usually do?"
2. Various types of interviews: "What do experts say they do?"
3. Contrived techniques: "What do they do when they are constrained in some way?"

A more recent taxonomy proposed by Dieste and Juristo (2011, p286) and based on that of Hoffman et al. (1995) has the following activities at the top of their hierarchy:

1. Introspection & observation
2. Interviews
3. Contrived techniques
4. Questionnaires
5. Picking from a list of attributes
6. Prototyping
7. Scenario analysis
8. Diagramming

Note that the first three (introspection & observation, interviews, and contrived techniques) correspond to the three categories from Hoffman et al. (1995). They also are broken down into more specific tasks by Dieste and Juristo (2011). The remaining five categories (questionnaires, picking from a list of attributes, prototyping, scenario analysis and diagramming) do not have any sub categories in the Dieste and Juristo (2011) article.

The choice of which technique to use is highly dependent on the context of the KA project, such as what needs to be elicited, who the experts are, what they are familiar with, etc. However, there has been mounting evidence that the first category "Analysis of the tasks that experts perform" is not as successful as the other two (Hoffman et al., 1995, p144). As for the distinction between interviews and contrived techniques, it is not quite as settled. Hoffman et al. (1995, p144) concludes that in many circumstances, contrived techniques are more beneficial. However, the more recent and comprehensive analysis performed by Dieste and

Juristo (2011, p299) came to the opposite conclusion, suggesting the evidence points to (structured) interviews performing better than contrived techniques in the majority of cases. It seems most likely that the actual finding of which tends to be more beneficial will depend on the domain being modelled.

Empirical research which makes use of KA tends to do so by combining multiple techniques from these papers. A typical example may be to use an unstructured interview to get a broad understanding of a domain and build an initial basic model, then follow up with more structured interviews in order to refine this model. Alternatively, one might start with individual interviews with multiple experts, then culminate in a group session with all experts to clarify any unresolved differences (as advised by McGraw and Seale, 1988).

As discussed previously, BNs are regularly constructed with KA. Of the BNs constructed via KA, they almost exclusively use interview techniques such as workshops or focus groups. This thesis focuses more on contrived techniques, delivered via surveys.

2.3.3. Reliability and Validity in Expert Judgement

Two key concepts in the field of KA and indeed most scientific endeavours, is that of reliability and validity.

> “[Reliability] concerns the extent to which an experiment, test, or any measuring procedure yields the same results on repeated trials” (Carmines and Zeller, 1979, p11)

Incorrect judgements can be made due to natural variance in the phenomenon being discussed (true variance) or by an error made by the expert answering questions, independently of the true variance (error variance). The more error variance there is, the less reliable an instrument is (Guilford, 1978). For example, consider multiple experts all asked to estimate the weight of a ball, each giving different responses in a seemingly random pattern. In this case, it is reasonable to assume there is *no* true variance, as the weight stays constant, and all of the differences can be explained solely by error variance. As such, asking the question to several experts is an *unreliable* instrument for weighing the ball.

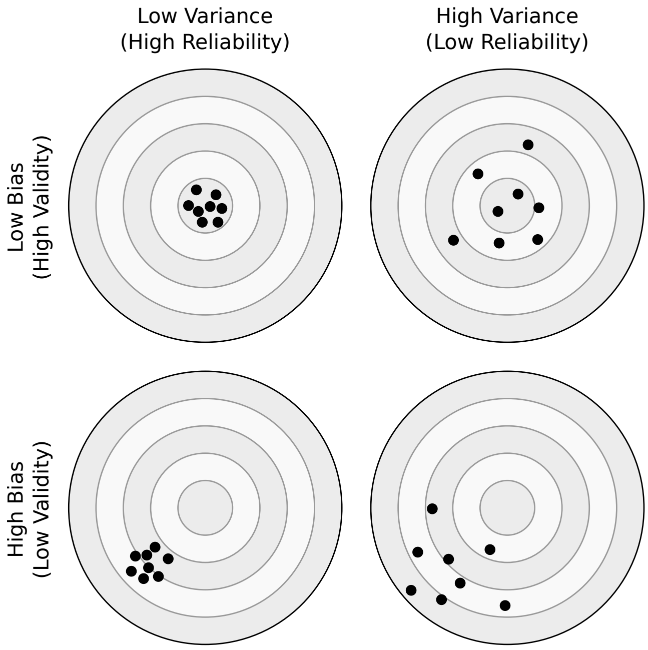


Figure 2.8.: Visualization of reliability vs validity: The centre of the target is the "true" value being measured with imperfect tools, and each imperfect measurement is a black dot on the target.

Validity is different to reliability, but equally important:

> "An indicator of some abstract concept is valid to the extent that it measures what it purports to measure." (Carmines and Zeller, 1979, p12)

If multiple experts answer a particular question by giving a similar judgement, however that judgement is incorrect, then it doesn't matter that it exhibits a high level of reliability. The fact that the question was answered incorrectly means that the judgements are not valid. This sometimes arises when a question is ambiguous or confusing, and the expert is unaware of exactly what is asked. In the above example of estimating a balls weight, perhaps the question didn't clarify which ball, or the ball was in packaging and it didn't clarify whether to include the weight of the packaging in the final estimation.

A common way to visualize the difference between reliability and variance is via a target (Figure 2.8), where each measurement is placed on the target and the centre is the hypothetical true measurement. From this it is clear that both reliability and validity are important, in order to make inferences about a true measurement when provided with many different measurements from experts.

2.3.4. Biases During KA

There are several biases which present themselves when an expert is asked to provide their judgment[5]. The result of systematic biases that present during KA is a decrease in reliability and/or validity. Thus, it is important to make efforts to mitigate any source of bias when conducting KA.

Early research by Tversky and Kahneman (1974) identified heuristics which are used by people when making judgments. These heuristics tend to result in systematic biases during certain elicitation tasks. To illustrate, two of these are highlighted below:

Anchoring Bias Anchoring refers to when somebody is unintentionally prompted with a figure before they are asked for their judgement on the matter. For example, Lichtenstein et al. (1978) asked two different groups to estimate the number of annual deaths due to various causes. Both groups were primed by being told a truthful value for a given cause of death. The first group was told 50,000 people die each year due to motor vehicle accidents while the second was told that 1,000 people die each year due to electrocution. Both were factually accurate statements, but people primed with the higher number gave higher estimates for other causes of death.

Availability Bias Availability bias can occur when the availability of information about one particular answer outweighs that of other answers. For example, it may be a topical matter and often appears in the media, or perhaps the expert has more experience with one. Tversky and Kahneman (1973) proposed several experiments to measure how availability can influence results. One asked participants to estimate the frequency with which the letter *k* (among others) appears in the first position of an English word, versus the third position. Despite it being more than twice times as likely to be in the

[5] Biases are often referred to as "effects" in the literature. e.g. Instead of an anchoring *bias*, it could be termed an anchoring *effect*.

> *third* position, respondents on average said that it is twice as likely to be at the *beginning* of the word. It was hypothesised that this is due to the fact it is easier for people to recall words beginning with *k*, rather than words where *k* is the third letter.

Since that early research by Tversky and Kahneman (1974), many other biases have been investigated. Arnott (2006, p60-61) presents a table of 37 different cognitive biases and relevant references.

Mitigating Biases (aka Debiasing)

One of the simplest ways to confront biases is the so called "consider the opposite" approach. With this, experts are encouraged to question their responses, by proposing situations in which they may be wrong. Fischoff (1981) showed that of 25 different debiasing techniques discussed, this was the most useful.

More recent research by Bazerman and Moore (2009) provide strategies for unfreezing, changing, then refreezing peoples intuitions to tackle the problem of biases. The idea is to convince people that despite past history showing that they may be pretty good at decision making, they are in fact subject to many of the biases discussed above (Larrick, 2004, p331). Once this has been shown to them (unfreezing), then steps can be taken to address the biases (changing) before encouraging them to maintain their new, good habits (refreezing).

2.3.5. Quantifying Expertise of Experts

In any discussion of expert elicitation, it is important to focus on what an expert is, and what qualifies somebody as an expert. Unfortunately this is repeatedly omitted in published articles. Expertise is often defined very loosely (Hoffman, 1998), and the difference between experts is rarely acknowledged.

> "In some domains it is difficult for non-experts to identify experts, and consequently researchers rely on peer-nominations by professionals in the same domain. However, people recognized by their peers as experts do not always display superior performance on domain-related tasks.

> Sometimes they are no better than novices even on tasks that are central to the expertise." (Ericsson, 2006, p4)

And as such, they encourage the authors of chapters in their handbook:

> "...to describe explicitly their empirical criteria for their key terms, such as "experts" and "expert performance". For example, the authors have been asked to report if the cited research findings involve experts identified by social criteria, criteria of lengthy domain-related experience, or criteria based on reproducibly superior performance on a particular set of tasks representative of the individuals' domain of expertise." (Ericsson, 2006, p4)

Discussions on experts qualifications are frequently found in literature on expert systems development. This is unsurprising, given that expert systems are "computer programs that exhibit some of the characteristics of expertise in human problem solving, most notably high levels of performance." (Buchanan et al., 2006, p87).

The question of who decides those that are experts has been addressed by some before. O'Leary et al. (2011) used a panel of meta-experts to help build a model for quantifying expertise in the field of taxonomists. This model was then used to weight opinions of experts during the knowledge elicitation phase. This is important because some experts will be experts in many areas of their field, while others may be highly specialised (O'Leary et al., 2011). If they are highly specialised, then they should only be considered an expert for a narrow subset of the questions which are likely to be asked of them.

Despite the idea of expertise being heavily studied, such discussions are lacking in the context of BN research. Martin et al. (2005) made use of experts to elicit information for a BN model of the effects of grazing on bird populations. They *did* make an effort to discuss the basis for choosing the experts in their study, referring to them as "experts with extensive experience in the response of birds to disturbance and field experience in grazed landscapes" (p268). However, they then *explicitly excluded* the prospect of distinguishing between levels of expertise, arguing that they are doing so to avoid "difficulties concerned with rating the comparative "accuracy" of each expert's opinion" (p269).

Another example of an attempt to quantify expertise is when eliciting BNs is Xiaoxuan et al. (2007). In this paper, they weight opinions of experts according to the criteria such as their professional title, self confidence of opinion, and proportion of their time they spend involved with the domain. However there is little to no discussion about how the weights were arrived at, or what happens if experts don't fit in one of the categories for a given criteria. Having said this, it is at least a starting point, showing that they are thinking critically about how much to trust each experts opinion.

2.4. Combining Expertise

For a long time, expert judgments have been combined together to form a single knowledge base. This has been performed in a wide variety of fields, including crowd sourcing (e.g Whitehill et al., 2009), expert elicitation (Onwuegbuzie et al., 2009) and classification problems (e.g. Seni and Elder, 2010). This section will highlight interesting work in each of these fields, in order to motivate the approach this thesis takes to combining multiple expert opinions into a single BN.

2.4.1. Combining Experts Opinions During Elicitation

A common example of expert elicitation that requires multiple experts to have their opinion combined is the focus group. This is where many experts discuss a common theme, and the person or people administering the group are responsible for mediating between experts to come up with a common solution. From a statistical perspective, Sniezek and Henry (1989) found that judgements formed by groups of experts were better than those obtained by averaging responses elicited from experts individually. This is contradicted by further research by Armstrong (2001) which found averaging is better than group consensus.

Some research making use of expert elicitation manually weights experts based on subjective measures of expertise. For example, Dransfeld et al. (2000) weighted experts by their time in an industry, position in a company, size and importance of the company, and a self judged ranking of expertise. A similar approach was taken by Xiao-xuan et al. (2007). This thesis discourages assigning weights to experts in such a manner, as it is quite subjective. It may be difficult to distinguish someone who is highly self confident but inaccurate from someone who is shy and lacks confidence, but is often more accurate.

It is only relatively recently that formal methodologies for combining expertise from focus groups have come about. Onwuegbuzie et al. (2009) present a formal methodology which explicitly takes into account factors such as which people were talking during a certain discussion, who was dominating the discussion, whether experts were in agreement, and how they showed their agreement (e.g. a nod of the head vs vocally).

2.4.2. Statistical Techniques for Combining Results

For quite some time, machine learning researchers have been interested in combining multiple models together to make more robust predictions. This process is referred to as ensemble learning, and its goal is to combine the results of multiple “weak learners” into one strong learner which is much more accurate. Early examples of this type of model include Boosting (Schapire, 1990) and Bagging (Breiman, 1996). The continual addition of new ensemble algorithms lead to more theoretical research on how best to combine results from multiple models. For example, Kittler et al. (1998) formalized many common combination rules, which are used to combine the results of multiple models.

In parallel to ensemble learning, another more recent field of research has also investigated statistical approaches to combining results from multiple sources. This is the field of crowd sourcing (Quinn and Bederson, 2011), which has gained a lot of attention due to the proliferation, cheap cost, and accessibility of online services for performing simple questionnaires, such as Amazon Mechanical Turk (AMT)[6]. In the field of crowd sourcing, participants are usually lay people rather than experts, and as such the research often has no knowledge about whether they are “good” at the task at hand or not. As such, there have been statistical techniques

[6] `https://www.mturk.com` (Last retrieved 2016-04-01)

proposed, which attempt to measure how expert a participant is, solely based on the way they answer questions.

The problem with using crowd sourced data is that the people involved in producing data are unknown to the researchers. They may be poor at the task at hand, or they may even be actively adversarial, intentionally selecting the incorrect answer. As a result, the data collected can be very noisy. Recent work in this field has resulted in several algorithms that are able to account for various sources of bias, in order to better discover the underlying ground truth from noisy data sets (e.g. Snow et al., 2008; Sheng et al., 2008; Whitehill et al., 2009; Raykar et al., 2010; Wauthier and Jordan, 2011; Organisciak et al., 2012; Zhou et al., 2012; Bachrach et al., 2012).

Majority Vote

A naive approach to deciding on the ground truth for a question is to take the majority vote for each question. If there are 10 responses to a question with three possible answers, the inferred correct answer would be the one with the most responses. This approach is often preferred for its simplicity (Lam and Suen, 1994), and it rapidly improves accuracy with the addition of more participants (Lam and Suen, 1994) although it saturates relatively quickly too (Bachrach et al., 2012). The majority vote has proven to be useful in many domains, and often provides equal or better results than more complex statistical or Bayesian approaches (Bachrach et al., 2012).

Expectation Maximization Algorithm

Although majority vote is simple and effective, there is a plethora of recent research investigating alternatives. Many of them are based on the early work to use the expectation maximization (EM) algorithm to combine multiple responses into a single model (Dawid and Skene, 1979). The intuition for this algorithm is that it would be nice to use the accuracy of each expert to be able to weight their responses[7]. Dawid and Skene (1979) opted to use the EM algorithm for this task, albeit in a clinical setting. They were interested in receiving multiple, noisy responses from a single patient, to questions posed by multiple different clinicians in order to obtain the ground truth response to each question. The intuition for EM is to analyse the experts responses, assuming everybody is 100% accurate. Once done, the majority vote can be used to estimate the ground truth for the question, and then experts accuracy is adjusted depending on whether their responses align with these ground truths. Once their accuracy has been calculated, the ground truth is recalculated, with each persons response being weighted according to the accuracy assigned in the previous step. These two steps are repeated until the ground truth converges.

More Advanced Crowd Sourcing Algorithms

Although Dawid and Skene (1979) first proposed EM for combining multiple opinions like this in the 1970s, it wasn't until the past decade that there has been a renewed push for research in this area. The motivation for Dawid & Skene was to assess individual patients who were interviewed by multiple clinicians on different occasions. These days, an increased thirst for data, and also online tools such as AMT have enabled the use of crowd sourcing as a low cost method for gathering and/or creating data. Examples of this are so called "citizen science" projects (e.g. Chklovski and Gil, 2005; Raykar et al., 2010). Instead of individual patients interviewed by multiple clinicians on different occasions, crowd sourcing has individual ground truths, as decided upon by multiple lay people on different occasions.

Some of these allow for incorporation of question difficulty (Whitehill et al., 2009) and active learning (Sheng et al., 2008; Wauthier and Jordan, 2011), resulting in improved results. Others allow calculation of which participants contributed the most to correct responses (Bachrach et al., 2012).

This thesis investigated applying the Majority Vote algorithm and the EM algorithm in order to collate survey responses into an authoritative BN structure

[7] One could equally replace the word accuracy with "correctness" or "expertise" or other measures of an experts "goodness", however accuracy is the term used in a lot of this type of research.

and parameters.

2.5. Surveys

Traditionally surveys have been used by demographers and sociologists, in order to study large populations of people while only having to sample a representative few (Babbie, 1990, p42). The goal is to study a broader population by taking a sample of them and asking them questions. If done correctly, it is possible to make certain assumptions about the entire population. In order to conduct a survey correctly and successfully, there are many considerations which need to be managed. The survey methodology has attracted a large amount of researchers since an early paper by Neyman (1934) discussed various sources of errors which can arise in survey data due to problems with sampling.

2.5.1. Total Survey Error

This section will discuss some facets of a *good* survey. The most common approach to this has been termed the "total survey error" (Andersen et al., 1979; Groves and Lyberg, 2010; Weisberg, 2009). This concept tries to enumerate possible sources of error which appear in the process of conducting a survey, from designing and preparing the survey, through to administering it and then analysing the data. Each of these stages is capable of introducing their own bias or variance which affect the conclusions of the survey in different ways, and minimizing each of these sources of error results in better surveys.

Groves and Lyberg (2010) summarise total survey error by grouping error sources into two main groups. The first are errors introduced by individual responses to questions by individual survey participants. Examples of these include:

- Questions in a survey that do not adequately measure the construct they were designed to measure.
- Errors or misunderstandings on behalf of the participant when answering questions.

- Processing errors caused by incorrect encoding or analysis of responses by the survey administrator.

The second source of errors arise when aggregating multiple responses from different participants and generalising the results to a broader population. Example of these are:

- Sampling errors resulting from the survey respondents not being representative of the total population to be measured.
- Non-response error where participants from the sample who failed to respond to the survey would have answered differently from those who did, biasing the results.

The goal of any survey is to collect reliable and valid measurements. This approach of enumerating each possible source of error and then trying to mitigate it is very similar to the concept of debiasing in KA projects, discussed in Section 2.3.3 (p44). Whether conducing a survey or a more traditional KA interview, reducing errors and biases is important.

2.5.2. Surveys as a Knowledge Acquisition Tool

Historically, surveys have been used to collect data about a sample of individuals in order to extrapolate to an entire population and make inferences. This is somewhat similar to trying to perform KA with a small number of experts, and treat their knowledge as a proxy for the ground truth about a particular domain. However, asking people questions in order to collect data for analysis is not limited to sociology research. Rather, surveys can also be used as a way to extract tacit knowledge from experts and make it explicit. In this way, they are a powerful KA technique, if used well.

It is possible to construct surveys so that the respondent needn't know the complexities of the model that is being built based on their responses. This is useful for BN elicitation because a survey should be able to be constructed such that the experts needn't be familiar with the inner workings of BNs or how to construct them. As Babbie (1990, p.228) said, "questions should make sense to respondents ... the most important implications of the questions might not be evident to them".

Relatively recently, it has become more common for surveys to be used as a KA tool to elicit knowledge, rather than to elicit opinions or demographic information in social surveys. This type of application of surveys is in its infancy, as evidenced by Baker et al. (2014) who conducted expert elicitation using face to face interviews and also web based surveys. Their results were inconclusive and suggested that further research is required to investigate whether surveys are a suitable replacement or not.

This thesis took an approach similar to Baker et al. (2014) in order to investigate the usage of surveys as a KA tool.

2.6. Chapter Summary

This literature review introduced BNs, why they are useful for solving problems, and how they are typically constructed. The review of BN literature covered various techniques for reducing the magnitude of the BN elicitation task. Some approaches focus on the structure elicitation, and others on the CPT elicitation. Both are important for reducing the total burden on experts when eliciting BNs. The review then looked at the field of KA in order to identify some of the short comings with traditional approaches to BN construction. The survey methodology and its relationship to KA was also investigated to set the stage for the following chapter to discuss the method by which this thesis applies the survey methodology to BN elicitation in order to resolve some of the issues described in this review.

3. Methodology

This research is framed using the Design Science research (DSR) methodology. This chapter begins by briefly introducing DSR in Section 3.1, followed by reiterating the problem this project aims to address. Following this is a detailed depiction of the propositions this research seeks to address in Section 3.3. This is then followed by introducing and discussing the two main research outputs to come from this research in Section 3.4. Finally, the study which took place to evaluate these propositions is discussed in Section 3.5.

3.1. The Design Science Research Methodology

When conducting research, particularly in Information Systems (IS), the outcome is often a man made artifact (Simon, 1996). Such examples might be a decision support system (e.g. Arnott, 2006), or processes for improving business management (e.g. Van Aken, 2005). This type of research which results in artifacts has been well studied by scholars interested in the science of design, and comes under the umbrella of the Design Science Research (DSR) methodology (Hevner et al., 2004; March and Smith, 1995). Design science is the "science of the artificial" (Simon, 1969), where researchers produce new artifacts and study their use, impact and implications. This is distinct from the so called *natural* science method which can be thought of as researchers investigating and explaining naturally occurring phenomenon (Walls et al., 2004, p45).

A recurring theme among many, but by no means all design science researchers is the importance of producing research that can be applied:

> "The conventional wisdom in the design science literature is to promote generality. This is assumed to foster research rigour and make theories relevant to a (broad) class of problems and future practice. At the same time, leaving considerations of design alternatives outside the scope of design theories may deny practitioners important guidance. There is a general trade-off between generality of a theory and its ability to account for specific issues." (Lukyanenko and Parsons, 2013, p168)

A similar sentiment is also echoed by the Information Systems Design Theory (ISDT, Walls et al. 1992, 2004) which encourages researchers to "go beyond descriptive and normative theories to provide specific guidance to the design process through a prescriptive mode" (Walls et al., 1992, p49). As such, this research maintains that a prescriptive approach should be offered as research output, rather than stopping at a general framework without much guidance in how to implement it as a workable software solution. The following section discusses the problem which was identified in Section 1.1, and for which this thesis presents a prescriptive framework for addressing.

3.2. Problem Identification

In the terminology of Peffers et al. (2007), *problem identification* is usually the first stage of any DSR project, or indeed any research project in general. For this project, the identified problem arose out of an increasing need to construct BN models by eliciting knowledge from experts. Existing interview based elicitation methods are exhaustive, but also exhausting (Hoffman et al., 1995, p134) both for the knowledge engineer and any experts participating. Thus, they are a contributing factor to the knowledge bottleneck faced when constructing models via expert elicitation (Hoffman et al., 1995, p134). The result of this is that less experts will be able to commit the required time to elicitation projects. Even if a large number of experts are able to commit the time required to elicit a BN, integrating the opinions of each expert into the final model is challenging, as methods for weighting and combining different judgments are required (Martin et al., 2012, p35). These problems were discussed in greater detail in Chapter 1 culminating in the following research question:

> *How can the process of eliciting knowledge for construction of Bayesian Networks be improved by making use of online surveys instead of face-to-face interviews?* (Section 1.2, p5)

This was elaborated on with reference to the following two sub questions:

1. *As more experts are consulted, how can the total time and effort involved in KA for BNs be reduced?*

2. *As more expert opinions are gathered, how can they be collated into a single BN model without significantly increasing the workload to resolve differences?*

Chapter 1 discussed the need for a new method of eliciting BNs which exhibited the following improvements over the existing method of face to face interviews:

- *Less time* required of experts and researchers alike.

- *Easier integration of differing opinions* into the BN model.

- *Less constraints* on where and when experts can contribute.

As such, this research is rooted in an applied problem, that of how to perform expert elicitation of BNs. The resulting solutions proposed by this research are built on work done by researchers studying the survey methodology (e.g. Babbie, 1990), traditional ideas about how to elicit BNs from experts (e.g. Kjærulff and Madsen, 2013), and also from the field of crowd sourcing (e.g. Quinn and Bederson, 2011). Gregor and Jones (2007) discussed the anatomy of a design theory, specifying eight components which should be present in any information systems design theory (ISDT). One of the important parts of an ISDT is *testable propositions* (Gregor and Jones, 2007, p327). This is similar to the more traditional natural science method, whereby a series of falsifiable hypotheses are announced before conducting any experimentation. The following section discusses a set of propositions that arose out of the research questions listed above, and which were addressed throughout this research.

3.3. Propositions

The testable propositions introduced in this section mostly relate to two main constructs: the *number of experts*, and the *time required of each expert.* By minimizing the time required of experts, it is anticipated that the number of experts can be maximized, and thus the final BN will include a more diverse range of knowledge. The propositions are geared towards verifying whether SEBN is indeed able to achieve this goal.

This section discusses a series of propositions designed to clarify the situations in which the survey based BN elicitation technique is expected to be suitable (Section 3.3.1), and those where it may not be suitable (Section 3.3.2). In addition to these two sections discussing the process of eliciting a BN, Section 3.3.3 discusses propositions about the resulting output of that process and the quality of the resulting BN.

The propositions below are the precursor to Section 3.5 which discusses the details of the evaluation that took place. The evaluation was conducted to support or disprove the propositions shown below, and verify whether the two constructs of interest .

3.3.1. Times When SEBN Would be More Suitable Than Traditional KEBN

SEBN was designed to reduce the burden on experts in order to encourage more to participate, compared to traditional KEBN. However, there will be some times where traditional KEBNwill still be more suitable than SEBN. This section documents the claims whereby it is proposed SEBN will be more suitable. Section 3.3.2 documents propositions relating to times when it may be less suitable.

The propositions laid out here are based on accepted properties of surveys, notably, how they compare to interviews as a data gathering technique. This comes from a wealth of literature in a range of disciplines such as psychology and sociology (see Section 2.5, p53). These fields have made use of questionnaires as a data gathering technique for a long time.

Proposition 1. *SEBN will require less time of each expert than traditional KEBN.*

The method proposed in this research outlines direct questions, compared to semi-structured, qualitative interviews. It also discusses how the questions can be divided up among a group of experts, so as to only ask a subset of all possible questions to any given expert. As a result of this, the total time spent responding to an online survey should be less than the time spent with an equivalent interview.

This is particularly important, because experts of the type desired for most knowledge elicitation tasks are often people high up in their field, who have many commitments. As such, their time is valuable, and attempts to reduce the burden on them should be investigated and considered.

Proposition 2. *SEBN will require less time from researcher than traditional KEBN.*

All elicitation techniques require some level of effort be expelled by the person conducting the elicitation. Both traditional KEBN and SEBN require initial planning on behalf of the researcher. This is spent deciding how the interview or survey should be laid out, what the goals are, and what questions will be asked to achieve those goals. When researchers choose to use surveys as a replacement for an interview, they can take a lot more effort to prepare. While it is true that interviews can take an equally long amount of time to prepare, semi-structured interviews provide more scope for exploration during the interview. In the survey case, the researcher must take extra care to ensure all desired avenues of discussion are included in the survey, as they are not able to ask any further clarifying questions once the surveys have been sent off. While this is still relevant for the survey method proposed in this research, the prescriptive nature of the technique reduces this burden by providing guidance about, and constraining the questions that should be asked.

When administering the questions, online surveys require much less of the researchers time than an interview would. Once the online survey is prepared, knowledge engineers need not spend time administering questions at all, until it comes time to analyse the results. This is in contrast to interviews, which can range from

	Preparation	Recruiting	Administration	Transcription	Analysis
Surveys	●●●	●●●	●○○	●○○	●○○
Interviews	●●○	●●○	●●●	●●●	●●●

Table 3.1.: Rough estimates of the relative time commitment required of the researcher for surveys and interviews. Estimates are indicative only, to show the occasions where surveys would be expected to take less of a researchers time.

single interviews over an hour or more, or indeed projects that span multiple interviews over several days/weeks/months (Milton, 2008, p50). There are private companies who are able to conduct interviews on behalf of a researcher, reducing the interviewing time of the researcher to virtually nothing (e.g. Survey Monkey[1], Lime Survey[2]). However, such companies tend not to specialise in knowledge elicitation of the type required for BN elicitation, but rather more general tasks such as collecting demographic data or conducting market research.

Not only does the researcher spend less time interviewing experts, they will also spend less time transcribing results. In an interview, the researcher usually records the interview, transcribes it, then analyses the transcription to extract relevant information (Milton, 2008, Chapter 4)[3]. With SEBN, the data is entered in a format that is already prepared for analysis. The transformation of information from expert knowledge into a functioning BN is conducted completely by software in the survey approach, whereas things such as resolving differences between experts must be done manually by the expert in the traditional KEBN approach.

A high level summary of the estimated amount of time required by experts in survey vs interview based methods is shown in Table 3.1.

Proposition 3. *SEBN will be more suitable than traditional KEBN with geographically dispersed experts.*

Although not always the case, sometimes the only experts which are available to

[1] `https://surveymonkey.com`

[2] `https://limesurvey.com`

[3] Note that this may sometimes be different in KEBN (compared to regular interviews), where BNs can be built during the interview by the researcher in conjunction with the experts, using commercial software such as Netica, BayesServer, etc.

contribute to a project are limited by the availability of those that the researchers are physically able to meet. Surveys solve this issue by being administered without the researcher being present. Therefore, researchers are able to mail out, or more recently, conduct online surveys to elicit information from experts. Having said this, it is also true that interviews can be conducted online or over the phone, although this does not seem to be a prevalent technique used in many knowledge engineering projects at this point in time.

Proposition 4. *SEBN will be better able to distinguish knowledgeable experts from others compared to traditional KEBN.*

The literature review in Section 2.3.5 (p47) discussed how there is a gap in the way in which many knowledge elicitation projects quantify expertise. The level of expertise is often gauged by the number of years they have worked or the position they hold (e.g. Xiao-xuan et al. (2007)). However, there are always going to be people who have worked for less time, or in a lesser position, but who would be considered "more expert" than somebody in a more senior position. Also, there are times when experts are chosen as a matter of convenience, and in fact are not as expert as one may hope.

The survey method explicitly defines a procedure for incorporating knowledge from experts of varying degrees of expertise. The techniques are adopted from the field of crowd sourcing (Quinn and Bederson, 2011), which has a history of successfully taking information from multiple respondents of unknown expertise, and weighting each respondent based on how often they tend to provide useful information in their answers[4]. When the problem is constrained even more to people who can reasonably be presumed to have expertise in a problem area, the same algorithms differentiate those with higher or lower levels of expertise.

[4] "Providing useful information in their answers" could equally be termed "answering correctly", but that becomes much more difficult to define when there is no gold standard answers to compare participants responses with.

3.3.2. Times When Traditional KEBN Would be More Suitable Than SEBN

Just as there are valid reasons to choose surveys in preference to interviews, there are also reasons to opt for interview techniques. This section introduces propositions relating to features of traditional KEBN, that are preferable compared to SEBN.

Proposition 5. *Traditional KEBN will be more flexible than SEBN.*

In the context of this research, "more flexible" refers to the ability to adapt the elicitation process in response to answers given during elicitation. Compared to questionnaires, interviews are able to further investigate responses given by experts[5]. This can result in new and unexpected knowledge that was not thought of before the interview took place. This is not the case with surveys, whereby the set of questions is fixed before the first survey is administered, and should not change during the entire elicitation process.

Proposition 6. *Traditional KEBN method will be more suitable than SEBN when only a small number of experts are available.*

The questionnaire method described in this thesis (Chapter 4 and Chapter 6) allocates a subset of question to each expert. Thus, if there are less experts available then more questions need to be allocated to each of them. This results in more time required of each expert (Figure 3.1) and the benefit of using SEBN over traditional KEBN diminishes. The time an expert spends answering the extra questions allocated to them will probably be better spent in an interview with the researcher. This allows all of the flexibility described in Proposition 5. The actual number of experts required to make SEBN a better use of experts time depends on the problem domain, specifically the number of variables and variable categories to be included in the BN.

[5] Note that, consistent with the rest of this thesis, the term "interview" is being used to refer to semi-structured interviews or focus groups. If the term was used to describe a survey administered face to face or over the phone, then they would probably exhibit the same lack of flexibility shown by online surveys.

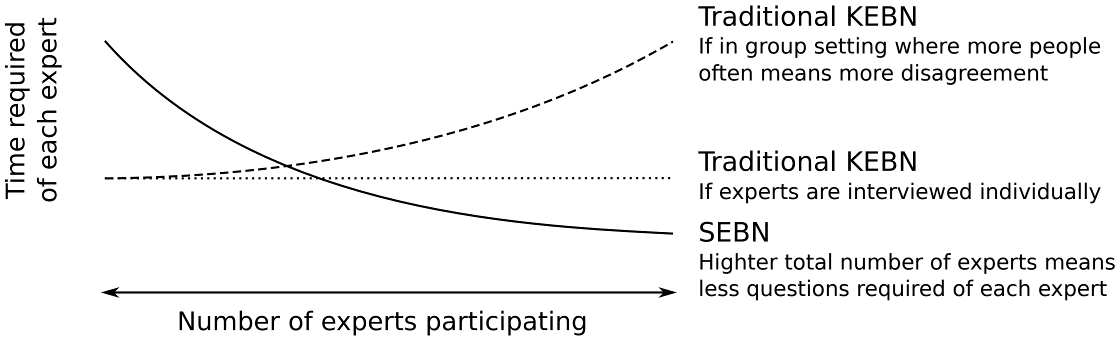


Figure 3.1.: As more experts participate in a survey, each is allocated less questions. Whereas with traditional KEBN, more experts in a group will likely increase the chance of disagreement which require resolution.

This is a particularly important facet of this thesis. The nature of many KA projects is that there is not a large pool of experts to draw on. Therefore, one must be cautious before deciding if a method such as SEBN is suitable, after investigating the number of experts who can make themselves available to participate in a particular project.

Examples of projects which likely have fewer experts to consult with include commercial projects that aim to build predictive models for the purpose of gaining a competitive advantage. When a commercial organisation invests in such a model, it is unlikely that many competing organisations would be willing to pool their respective experts together in order to build such models. However, in other fields such as government, not-for-profit organisations, or research institutions, there should be more incentive for a larger, more diverse group of experts to collaborate in constructing models using SEBN. It is anticipated that such fields would be able to benefit more from an approach such as SEBN which depends on the availability of a larger number of experts.

3.3.3. Quality of BNs Elicited Using SEBN vs Traditional KEBN

Whereas the two previous sections presented propositions discussing the merits of the *process* of eliciting BNs using SEBN or traditional KEBN, this section discusses the actual output of the processes - the BNs themselves. For the purpose of this research, “quality” was defined as the ability of a BN to model a specific probability distribution accurately. As a BN is a model, and all models are simplifications on some level, a worse BN model is less able to represent a desirable probability distribution. In the pathological case, almost zero time is required to randomly generate a BN, however that would not be very interesting or useful.

The following propositions all reference the idea of two networks using the same variables, one elicited using SEBN, and the other elicited using traditional KEBN. This concept is discussed in much greater detail in Section 3.5.

Proposition 7. *BNs elicited using SEBN and traditional KEBN will not be identical.*

Given that the same experts are not likely to produce the same BN twice using traditional KEBN, it is even less likely that two disparate techniques would produce an identical network. This is due to natural uncertainty inherent in the domain as well as experts ability to convey their knowledge consistently.

Proposition 8. *BNs elicited with SEBN and traditional KEBN will be* similar *in structure.*

Although Proposition 7 discussed why it is unlikely for two elicitation techniques to produce *identical* networks, there should still be similarities. If both networks are elicited from experts, and the arcs are meant to represent causal relationships, then the omission or addition of a causal relationship in one model provides feedback about how similar (or dissimilar) the models are. It would be ideal if the structure of a BN elicited using SEBN was similar to an equivalent BN elicited using traditional KEBN. Alternatively, if the structure was completely different and no arcs were the same, then that would provide evidence that SEBN tends to force people to think in different, sub-optimal, or incorrect, ways.

Proposition 9. *BNs elicited with SEBN and traditional KEBN will encode a* similar *probability distribution.*

Regardless of the method of elicitation, the goal is the same. That is, encode the probability distribution of some process or data into a BN that best matches that distribution. Thus, this proposition states that even if SEBN produces networks with different structures, they should be able to model the same probability distribution.

To address these propositions, this thesis produced and evaluated two artifacts which are discussed in the following section. The propositions are revisited in Section 8.2 (p195) to show how successfully this thesis addressed them.

3.4. Artifacts Produced From This Research

One of the core differences between *design* science and *natural* science is the focus on "produc[ing] a viable artifact in the form of a construct, a model, a method, or an instantiation." (Hevner et al., 2004, p82). This section discusses the two artifacts that arose from this research (Figure 3.2).

3.4.1. Method to Elicit BNs Using Online Surveys

The main artifact produced by this research is a new method of BN elicitation using online surveys (SEBN). March and Smith (1995) describe *methods* as:

> "...a set of steps (an algorithm or guideline) used to perform a task." (p257)

The method artifact produced by this thesis is the set of steps required to elicit BNs using online surveys. The goal was to propose a method that addresses the problem identified in Section 3.2 and adheres to the propositions in Section 3.3. The design and development of this process is described thoroughly and prescriptively in Chapter 4 and Chapter 6.

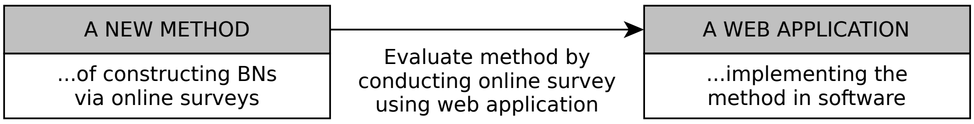

A NEW METHOD
...of constructing BNs via online surveys
Evaluate method by conducting online survey using web application
A WEB APPLICATION
...implementing the method in software

Figure 3.2.: The two main artifacts produced as a result of this research.

3.4.2. Open Source Web Survey Application

In order to be able to verify claims about the ability of the new elicitation method to produce BNs, it was implemented as an open source web application which can be deployed to administer online surveys for eliciting BNs (Serwylo, 2013). The DSR literature refers to such implementations as *instantiations*, which March and Smith (1995, p258) define as:

> "...the realization of an artifact in its environment." (p258)

Instantiations are helpful for evaluating whether proposed methods are indeed effective at solving the problem they were designed to solve. Also, by making software artifacts available for others to use and build upon, they can be used by practitioners in relevant fields.

The technical details of the web application are highlighted in Appendix A (p237). These details include the technology stack used to build and deploy the software, decisions that went into its design, screenshots, and documentation on how to configure it in order to conduct surveys.

This research continually evaluated the web application as it was being built. However, given the scope of this project, these evaluations were not as comprehensive as those which designed to evaluate the method to elicit BNs using online surveys. Section 8.7.2 (p215) discusses future research which should focus on formally evaluating the web application.

3.4.3. Note on the Terminology of "Instantiation"

Most DSR researchers use the term "instantiation" for concrete realisations of frameworks or workflows in software. The remainder of this thesis opts for the term "implementation", as it is a term more commonly employed by software developers to refer to the realisation of an abstract concept in a tangible piece of software (e.g. "Can you please *implement* this specification"). To ensure previous DSR publications are not misquoted, "instantiation" may still be used when referring directly to past DSR publications which use that term.

3.5. Evaluation

Despite early DSR researchers proclaiming the importance of evaluation (e.g. Hevner et al., 2004; March and Smith, 1995; Nunamaker Jr and Chen, 1990) it is only comparatively recently that researchers have presented specific theories of *how* to evaluate DSR research. Venable et al. (2012) and their predecessor Pries-Heje et al. (2008) proposed a taxonomy with which to organise various evaluation tasks. It is based on two dimensions: *naturalistic* ↔ *artificial*, and *ex-ante* ↔ *ex-post*. Naturalistic evaluations are those which take place in the setting the resulting artifact is designed to be used in, whereas artificial evaluations take place in more contrived situations such as laboratory experiments. Ex-ante evaluations take place during the research process and obtain results that inform further development of the artifact. Ex-post evaluations are conducted after the artifact is complete, and evaluate whether the artifact performs as expected, or they identify future areas of improvement.

The following section discusses two ex-post, naturalistic studies which were conducted in order to address the propositions from Section 3.3 and to provide further insight into how traditional KEBN and SEBN compare. This is followed by a discussion of the artificial, ex-ante evaluations that took place during the research.

3.5.1. Empirical Evaluation of SEBN

To satisfy the goals above, this thesis conducted an empirical evaluation of SEBN to investigate the propositions outlined in Section 3.3. This section provides a brief overview of the evaluation, to be elaborated on in Chapter 5 and Chapter 7.

To empirically evaluate SEBN, 107 participants took part in two online surveys, of which 67 participants completed. The first survey was to elicit the structure of a BN, and the second was to elicit the parameters of a known BN structure. The resulting BN structure and parameters were compared to a known, published, gold standard network to measure whether the elicited network is “good” or not[6].

[6] The evaluation in Chapter 5 (p103) discusses in great detail what it means to be a “good” network.

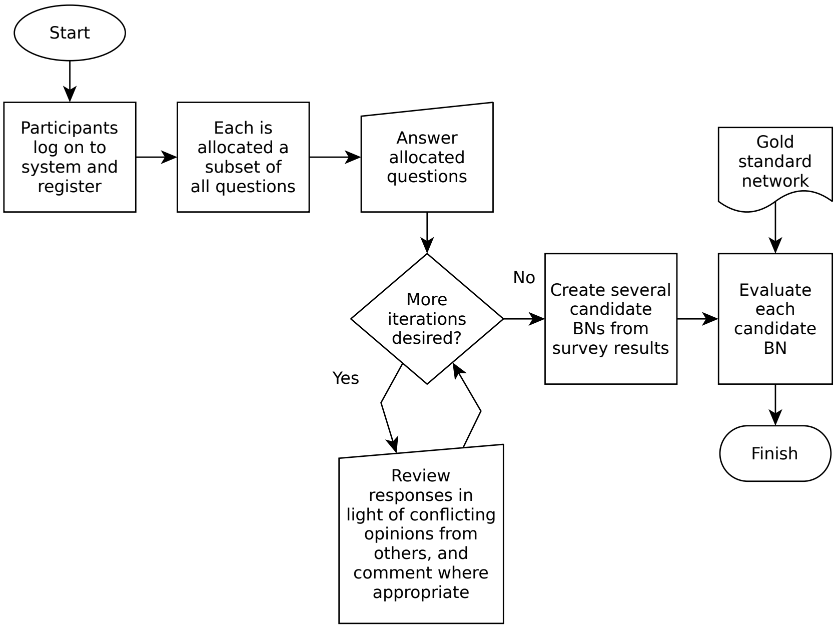

Start
Participants log on to system and register
Each is allocated a subset of all questions
Answer allocated questions
More iterations desired?
No
Yes
Review responses in light of conflicting opinions from others, and comment where appropriate
Create several candidate BNs from survey results
Gold standard network
Evaluate each candidate BN
Finish

Figure 3.3.: A broad overview of the evaluation process, whereby a survey is conducted, and the resulting BN(s) are then compared to an existing, gold standard to see how closely they match.

In addition, this evaluation serves to collect data about how long experts spent answering questions, how flexible SEBN was, and other data to help address the propositions. Figure 3.3 shows an overview of the evaluation process.

Given the scope of this research project, some trade-offs were made when designing the evaluation. The remainder of this section discusses what an idealistic experimental evaluation would entail so that it can highlight specifically where the evaluation conducted by this project deviated, and the reasons why. This research suggests that there are three criteria that should be met before comparing the structure and parameters of two BNs, such that the comparison is as meaningful as possible:

1. Both should consist of the same variables
2. Both should be constructed using the same technique (e.g. same learning algorithm or similar traditional KEBN procedure)

3. Both should make use of the same experts and/or data

Criterion 1 can be met by utilising an existing network that is already published. This was used to construct relevant survey questions, as prescribed by the survey method. Although some of the variables may not end up in the final network (if experts answering the survey don't think they are relevant), it gives the best basis for comparing two networks that are supposed to represent the same probability distribution. This is the approach taken by the evaluation in this research.

Given that this project proposes a new methodology for eliciting BNs, it is immediately obvious that criterion 2 is unsatisfiable. The closest to this is to compare with an existing network created using a technique that is *similar* to the one being proposed. Even in the absence of this, the comparison between BNs is meaningful, as it shows whether the networks both approximate the same probability distribution. The gold standard network used in the evaluation had its structure elicited in a way similar to traditional KEBN, which is preferable to a BN whereby the structure was obtained from data driven approaches.

Finally, for criterion 3, ideally the same experts would be used in the construction of the original network. However this is not always possible, due to them not being available, not being interested, not being identifiable, or simply not existing (e.g. if the network was learnt from data). Indeed, the evaluation uses a network which was produced in 1997, in another country. For these reasons this evaluation does not compare networks produced by the same set of experts.

Despite these issues that arise when trying to compare BNs from different sources, it is something that has been performed in the past to evaluate new BN construction techniques (e.g. Kennett et al., 2001; Tsamardinos et al., 2006).

3.5.2. Iterative Evaluation During Research

The previous section discussed the two specific studies conducted to evaluate the artifacts arising from this research. Given that DSR is an inherently iterative process (Hevner et al., 2004, p88), continual ex-ante artificial evaluations took place as the artifacts were conceived, designed, and implemented. These evaluations aided in the *formative design process* described by Gregor and Hevner (2013) as:

> "...an iterative design with intermediate test stages, where the testing and evaluation is formative, part of the development process, and is likely to include basic tests of validity using test data, scenarios, and simple experimentation" (Gregor and Hevner, 2013, p350)

In addition, it was anticipated that areas of improvement would be identified during the final two evaluations described in Section 3.5.1. Sources of improvement came from the researcher overseeing the survey, the feedback of participants, and logs gathered from the software which facilitated the survey. Given that the two survey evaluations formed the final iteration of this research, some of the identified improvements are discussed in more detail in the concluding chapter (p214).

3.6. Chapter Summary

This chapter provided a detailed discussion about the methodology used while conducting this research. The method stems from the DSR discipline, and is informed by many key articles in this field. The thesis produced two main outputs:

1. A method for eliciting BNs using surveys (SEBN).
2. An implementation of SEBN as an open source web application.

The method is be of use to technically minded people who wish to implement it for themselves, and is discussed in great detail in Chapter 4 and Chapter 6. The implementation is of more use to people wishing to elicit a BN but do not have a reason to create their own implementation from scratch. The evaluation is presented in Chapter 5 and Chapter 6 covering where SEBN excels, and where it is not as useful. It does this by addressing each of the propositions presented in Section 3.3 using the evaluation method described in Section 3.5. Those same propositions are then revisited in the concluding chapter in Section 8.2 (p195).

4. Building BN Structure Through Survey Based Elicitation

This chapter will discuss the first part of the main contribution of this thesis: a knowledge elicitation process for eliciting BN structures using online surveys (SEBN). The second aspect of SEBN which presents a method for eliciting BN *probabilities* is discussed in Chapter 6. Section 4.1 will start with a simplified overview of the process, which subsequent sections will build on.

4.1. Overview

Figure 4.1 presents a flow chart illustrating a simplified overview of SEBN. This section will briefly describe the main parts of the flowchart, and the intuition behind them. As with most surveys, this chapter is interested in deciding what the subject of the questions should be (obtain variables), producing the most appropriate questions (generate questions), combining multiple different expert responses to each question (collate answers), and then using these results to solve a problem (produce BN structure). The remaining parts of the process will be discussed in later sections of this chapter.

Obtain Variables Chapter 2 mentioned that the first step in constructing a BN is to elicit the variables of interest and their respective states. This is required because the structure is entirely defined by these variables and the relationships between them. Although proposing an approach for identifying variables is beyond the scope of this thesis, the future work chapter covers how the techniques proposed

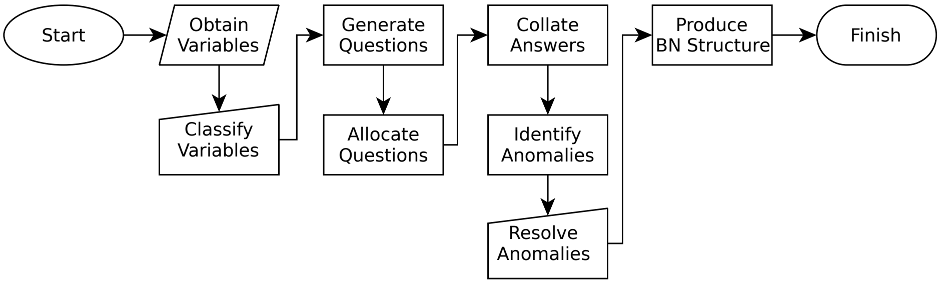


Figure 4.1.: Basic overview of the survey process. Each stage will be elaborated on in greater detail as this chapter progresses.

in Chapter 4 and Chapter 6 could be adapted to also facilitate the identification of variables (Section 8.7.4, p216).

Generate Questions The most basic of questions that is asked in order to produce a BN structure is: "Does variable X influence variable Y?". Although BNs need not only encode causal relationships, it has often been discussed that asking causal questions of experts enables them to use their basic intuition of causality to produce the BNs structure (Korb and Nicholson, 2011). The alternative is to train experts in conditional independence, and other nuances required for that.

The process of turning a list of variables into meaningful questions is discussed in detail in Section 4.2. This will be supplemented by Section 4.3 and Section 4.4, which focus on reducing the number of questions which need to be answered for the survey to produce meaningful results, but approach the problem from two different angles.

Collate Answers Once multiple experts have completed the survey, answering questions about causal relationships between variables, it is time to make sense of the data. The data which will be available is a set of explicit relationships that various people think exist. The main requirement is to consider how many people need to agree on a relationship before it is included in the final structure. Additional issues worthy of consideration include dealing with conflicting opinions, if somebody is more knowledgeable than others, and the accidental introduction of undesirable network structures when combining several disparate responses into a single BN. Section 4.5 introduces algorithms from the field of crowd sourcing, and discusses how one in particular can be used to help answer many of the issues raised above.

Produce BN Structure Once the answers are collated, it is a matter of outputting them to a format that can be understood by BN software. This could be done manually, or, if the survey was conducted online, automatically using the software used to conduct the survey. After collating the questions, but before producing a final BN, the resulting network structure should be analysed for anomalies. Numerous different types of anomalies, some of which are fatal to the success of a BN, others which are "nice to have" properties of a BN, are discussed at length in Section 4.6. Subsequently, Section 4.7 discusses the mechanics behind exporting BN structure to a format understood by major BN software vendors.

4.2. Generating Questions

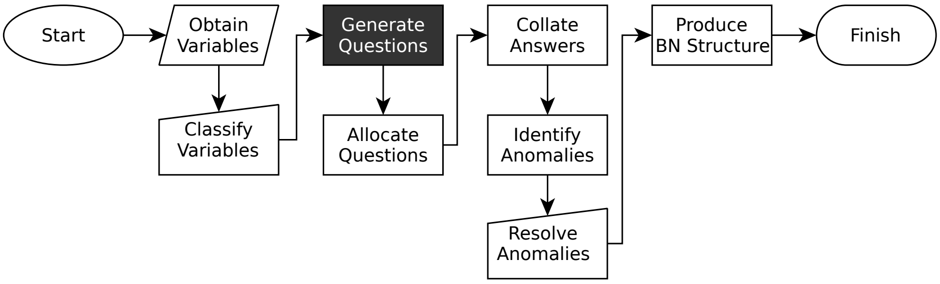
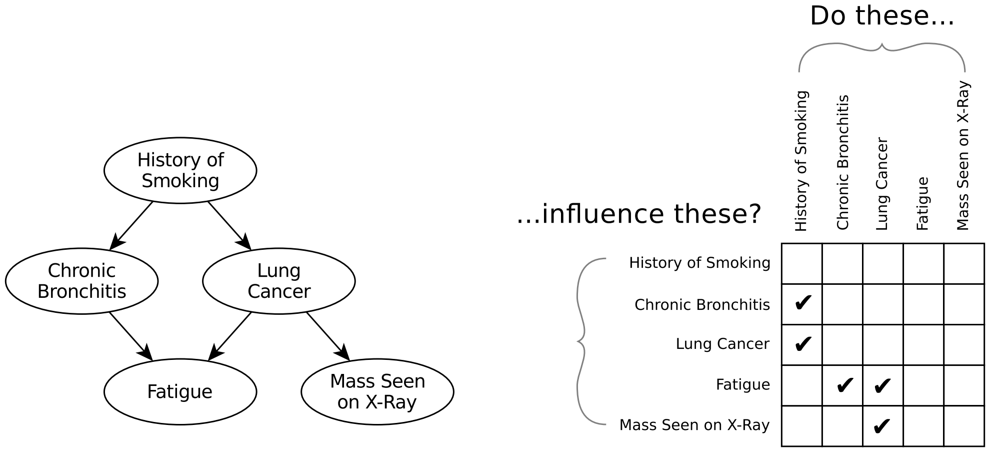


Figure 4.2.: Procedural generation of questions for the survey. Note that although "classify variables" is conducted prior to "generate questions" in the flowchart, it makes more sense for the purpose of explaining the survey technique to discuss question generation first.

The very first stage of structure elicitation in the survey process is to be able to generate a list of relevant questions. This section will begin with a naive algorithm for creating questions, and then augment and improve it as further details are discussed. The initial algorithm to generate questions is described in Section 4.2.1. This is followed by a discussion of how the variables should be labelled for maximum readability in Section 4.2.2. Following this, Section 4.2.3 mentions the need



	History of Smoking	Chronic Bronchitis	Lung Cancer	Fatigue	Mass Seen on X-Ray
History of Smoking					
Chronic Bronchitis	✔				
Lung Cancer	✔				
Fatigue		✔	✔		
Mass Seen on X-Ray			✔		

(a) A simple Cancer BN with five nodes and five arcs (from Cooper, 1999).

(b) Matrix encoding directed relationships in the Cancer BN from Figure 4.3a.

Figure 4.3.: Two different ways to visualise the relationships between variables - as a network and as a matrix.

for manual review of the questions, in order to verify that they are sensible. Finally, Section 4.2.4 talks about some issues which may arise after the questions are generated, and suggested techniques for combating these.

4.2.1. Initial Basis of All Questions

According to Proposition 2 (p61), the amount of work required of the knowledge engineer should be minimized. This particularly comes into play when the questions for the survey are generated. Thus where possible the questions should be generated procedurally. This is done by using an $n \times n$ adjacency matrix, where n is the number of variables in the network (Section 2.2.4, p33). Each entry in the matrix corresponds to an arc in the graph. This is shown visually in Figure 4.3, whereby the directed arcs in Figure 4.3a are encoded as entries in the adjacency matrix in Figure 4.3b.

To construct a BN based on a set of variables, it suffices to iterate over each cell in the adjacency matrix and ask if there should be an arc between the two variables in the resulting BN. However, this depends on the expert having an understanding about BNs and what it means to include an arc in that network between two variables. That is, each arc represents conditional dependence between two variables.

In order to circumvent the requirement for experts to understand BNs, this workflow makes the simplifying assumption that each arc in a BN represents a causal relationship. This assumption has been found to hold true in many applications (Heckerman, 1997; Spirtes et al., 2000). It may now be apparent that given a set of variables, the structure of a causal BN can be derived by iterating over every cell in the matrix, and asking:

> "Does X influence Y?"

Once each of these has been answered, the structure of the network is known. The remainder of this section will discuss how to refine this process, so that the questions are actually readable and provide enough context for an expert to answer in a survey context, where they may not be able to ask the knowledge engineer to clarify. Subsequent sections will deal with how to reduce the magnitude of the problem, so that less than n^2 questions are required.

4.2.2. Making Survey Questions Easy to Understand

For historical or technical reasons, a lot of commercial BN construction software (and indeed more general statistical software packages) require variable names to be alpha-numeric. As such, variables such as "Car Cost" have their whitespace removed and end up as the slightly less meaningful "CarCost". The side effect is that many of these software packages have variable *names* (machine readable) and also variable *labels* (human readable).

The readable labels required for this algorithm are even more human readable though. Rather than just adding whitespace or special characters where required, they need to be able to help provide context when used as a sentence fragment such as the X in "Does X influence Y?". For example, instead of changing "CarCost" to "Car Cost", it may be preferable to opt for something like "the cost of a persons car". Note how this fits much better when substituted into a more generic sentence:

- "Does CarCost influence ...?"

- "Does Car Cost influence ...?"
- "Does the cost of a persons car influence ...?"

Also note that this is not quite the same as a full blown description, which is also ideal. That is, if a user cannot make sense of a particular question, they should be offered some contextual help about the in-depth meaning of variables, beyond what the readable label has to say. In the BNE software, descriptions are shown as contextual help which is accessed by clicking question mark icons next to the relevant variables.

Algorithm 4.1 Basic algorithm to generate questions of the form "Does X influence Y?"

```
$\mathbf{V} \leftarrow$ Set of all variables
$\mathbf{Q} \leftarrow \varnothing$                                ▷ Generated Questions

for all $\mathbf{V}$ as $x$ do
    $\mathbf{x} \leftarrow$ReadableLabel($x$)
    for all $\mathbf{V} \setminus \{x\}$ as $y$ do
        $\mathbf{y} \leftarrow$ReadableLabel($y$)
        $\mathbf{q} \leftarrow$ "Does $\mathbf{x}$ influence $\mathbf{y}$?"
        $\mathbf{Q} \leftarrow \mathbf{Q} \cup \{\mathbf{q}\}$
    end for
end for
```

4.2.3. Manual Intervention and Rewording of Questions

As discussed in the previous section, it is worthwhile to automate the process of question generation as much as possible. However, there will likely be some cases where procedurally generated question are generated that turn out to be nonsensical and rather confusing. Therefore, Algorithm 4.2 also proposes that each question be read by the knowledge engineer to double check for clarity. It may be that in the future, after further refinement and evaluation of the survey method, this is a non-issue and can be skipped. But at this early stage in the development of SEBN, it is prudent to verify this.

During testing, one thing which came up was the difference between variable tenses. That is, one variable may be about the present tense, and another is about the future tense. Consider the following example of two variables, *Age (age of client)* and *Theft (car gets stolen)*, which when combined procedurally result in a question that doesn't quite make sense:

> "Does *age of client* influence *car gets stolen*?"

In this example, there are two variables, *age of client* (nominal variable, about the current state of a client) and *car gets stolen* (boolean variable, about a possible future event). Due to the differing tenses, the question is quite jarring. In such a case, manual intervention will be required by the survey administrator, to change it to something more suitable, such as:

> "Does *the clients age* influence *the chance of their car being stolen*?"

Notice how the use of the word "their" in this example depends on the variable being substituted at the start. In this case, it refers to information about "the client", but it may equally refer to information about their car. This would completely change the context of "the chance of their car being stolen" and make the question grammatically incorrect. Empirically, this seems to be more of an issue when background variables are related to problem variables (see Section 4.3, Figure 4.5).

4.2.4. Dealing with Questions that Don't Make Sense

Even after correctly semantically labelling variables (Section 4.2.2) and manually rewording nonsensical questions (Section 4.2.3), it is possible to produce some questions that simply don't make sense to some people. One expert may believe that a question is clear and unambiguous, whereas somebody else may think it doesn't make any sense at all.

A rule of thumb could be that if a relationship is confusing enough that people are unable to make sense of it, then it may not be a causal relationship worthy of inclusion in a model. In such cases it may be worth encouraging participants during the survey to answer "No" to questions "Does X influence Y?" if it doesn't make sense to them. Sometimes knowledge engineers make decisions to exclude information from models, if that information will cause confusion in the future. Modelling is a balancing act between being producing models that are understandable, maintainable, and simple, versus those that are the most faithful representation of reality that they can be.

Algorithm 4.2 Adaptation of Algorithm 4.1 which takes into account manual rewording of questions of the form "Does X influence Y?"

```
V ← Set of all variables
Q ← ∅                                          ▷ Generated Questions

for all V as $x$ do
    x ←ReadableLabel($x$)
    for all V \ {x} as $y$ do
        y ←ReadableLabel($y$)
        q ← "Does x influence y?"
        if NotSensible(q) then
            q ←ManuallyRewordQuestion($x, y$)
        end if
        Q ← Q ∪ {q}
    end for
end for
```

4.3. Constraining Possible Questions

One of the most immediate problems that arises from Algorithm 4.2 is the large number of questions that are required to be answered. Given Proposition 1 (p61) which states "Questionnaire method will require less time of each expert", this is not ideal. The process of generating questions did not describe any means of reducing the number of questions below $n^2 - n$ (where n is the number of variables). The total number of questions required quickly becomes unmanageable, thus negating some of the benefits of using SEBN over traditional KEBN, namely the reduced time required of each of the experts.

Previous research to elicit BNs via surveys (e.g. Xiao-xuan et al., 2007) or which used an $n \times n$ matrix to elicit information about a BN (Flores et al., 2011) did not sufficiently address the n^2 problem. Both asked experts to answer questions of the form "Does X influence Y?" for every combination of variables, with little

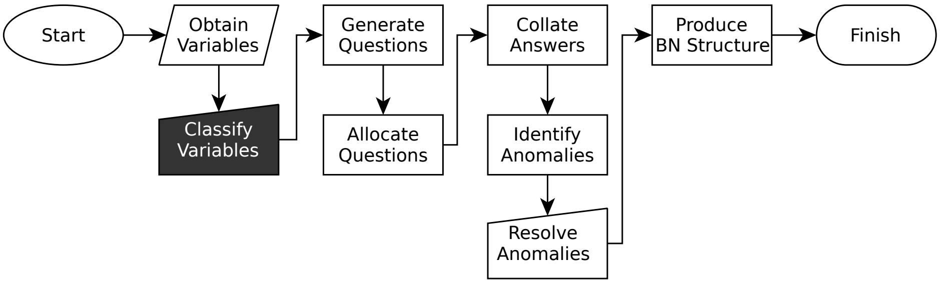


Figure 4.4.: Before generating questions, variables are classified into categories. This enables the total number of questions to be reduced.

to no discussion on how to reduce the number of questions before presenting them to the experts. This may be because these two papers were one off studies where the goal was to build a BN to solve a problem, rather than propose a more general approach to survey elicitation which is the goal of this thesis. To address this, Figure 4.4 amends the flowchart in order to facilitate constraints on the overall number of questions that need to be generated.

SEBN uses an approach based on that of Kjærulff and Madsen (2013, p152-154) which describes general classes of variables which tend to have certain causal dependencies among themselves. Figure 4.5 shows the four variable classes described, and how they are able to influence each other. For example, problem variables can influence symptom variables, but symptom variables never influence problem variables. An example of how this might be applied to the cancer network is shown in Figure 4.6.

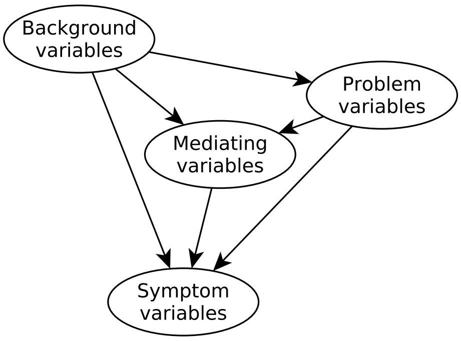
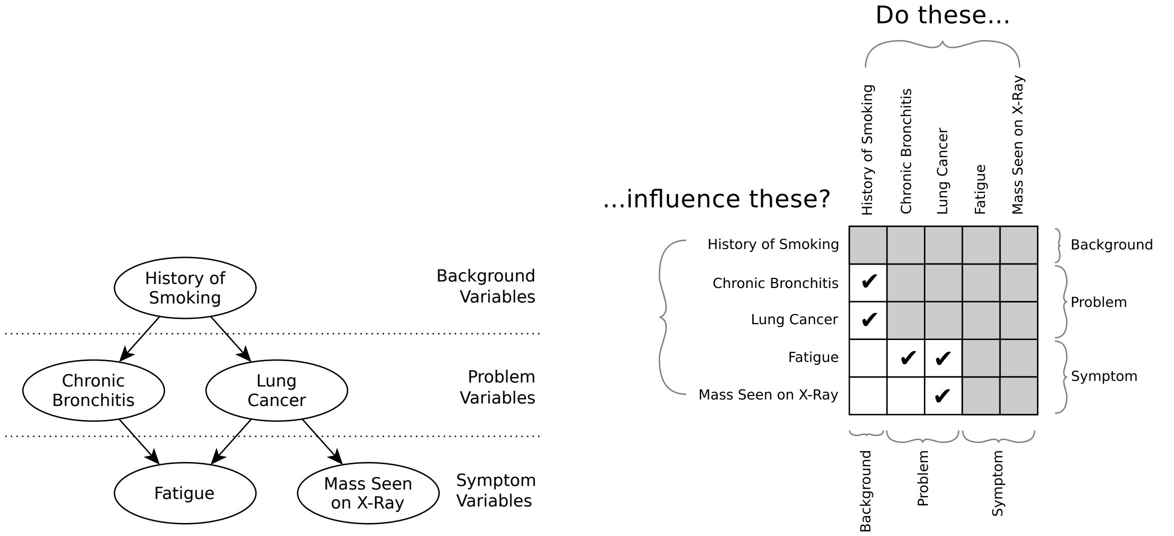


Figure 4.5.: Four general classes of variable and the logical dependencies between, them proposed by Kjærulff and Madsen (2013, p152-154).

Although there are some cases when the rules might be too rigid, applying this general principle greatly reduces the total number of possible questions which can be asked. This trade-off is acceptable, given that a model is by definition a simplification or abstraction of a real phenomenon. Note that applying constraints based on variable classes only works given that this thesis assumes the BNs being elicited represent causal relationships between variables. If non-causal relationships were allowed then such a constraint on questions could not be applied.



	History of Smoking	Chronic Bronchitis	Lung Cancer	Fatigue	Mass Seen on X-Ray
History of Smoking	▨	▨	▨	▨	▨
Chronic Bronchitis	✔	▨	▨	▨	▨
Lung Cancer	✔	▨	▨	▨	▨
Fatigue		✔	✔	▨	▨
Mass Seen on X-Ray			✔	▨	▨

(a) The classes of each variable in the cancer network.

(b) Greyed out squares are those which are ineligible due to inter-class constraints.

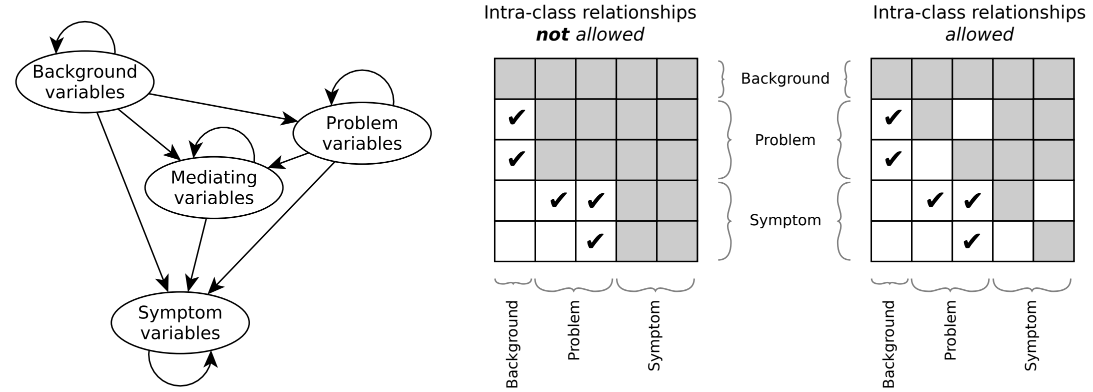
Figure 4.6.: Illustration of the variable classification scheme proposed by Kjærulff and Madsen (2013). Note how the number of possible relationships in the matrix is greatly reduced, thus reducing the number of questions required by a survey.

There are many ways in which this could be augmented to make it more flexible. The following sections describes two ways in which this could be adopted to constrain questions in different ways. This is followed by some considerations that should be taken into account.

Allowing Intra-class Relationships Figure 4.7 loosens the constraints of Figure 4.5, allowing variables to be able to causally influence other variables of the same class.[1]

For example, the car insurance network from Binder et al. (1997) might classify *Age* and *Has Advanced Driver Training* as background variables, and *Accident* as a problem variable. This is a good start, because the fact a person had an *Accident* cannot influence their *Age*, but their *Age* can influence their chance of having an *Accident.* If the model was constrained to the relationships depicted in Figure 4.5 though, then *Age* would not be able to influence *Has Advanced Driver Training.*

By default, the BNE software allows intra-class relationships. However, this can be changed by configuring the software appropriately.



(a) Elaboration of Figure 4.5 whereby variables within a particular class are able to influence each other.

(b) Number of questions required increases marginally as intra-class relationships are allowed. Note that even though the diagonal cells represent intra-class relationships, it is impossible for one variable to depend on itself for the purpose of a BN (or any DAG, for that matter).

Figure 4.7.: Relaxing constraints on the variable classes, by allowing intra-class relationships.

Adapting Categories for New Domains Described above is a general approach to classifying variables, which will help reduce the number of questions for most

[1] Note how Figure 4.7b seems to show that background variables cannot influence other background variables, as they are greyed out. However, this is only because the cancer network (Figure 4.6) only has one background variable. Given this variable can't influence itself, it is greyed out. If there were more background variables, then they would indeed be able to influence each other and this would be reflected in the adjacency matrix.

BNs. Alternatively, it may be more suitable to devise different categories for a specific domain which still impose a logical dependency. For example, the water management BN by Chan et al. (2010) (produced using traditional KEBN) has six distinct groups of variables:

1. Water Quality
2. Human Activities
3. Affordability
4. Management
5. Climate
6. Sustainability

These tend to have a logical dependency which helps segment the variables. For example, neither Water Quality, Human Activities, Affordability, Management, or Sustainability is able to influence the Climate for the purposes of this model. If SEBN was used, the number of questions which required answers would be reduced significantly using these dependencies.

Appendix E explores domain specific classes in more detail in the context of the Chan et al. (2010) BN. It shows that, similar to the generic classes outlined earlier, the number of survey questions which require answers is reduced by 50% from the naive case where any variable can influence any other. Although the reduction in questions using the classes outlined Appendix E is about the same as the generic classes above, it has the added benefit of being able to allocate questions from classes with well defined, domain specific semantics, to experts with particular knowledge in that area (see Section 4.4.2).

Who Assigns the Variables to Categories? In principle, classifying variables into background, problem, mediating and symptom classes requires somebody with a level of expertise. This is not dissimilar to some algorithms for learning BNs from data, where there is often a requirement that variables be ordered (e.g. the K2 algorithm, Cooper and Herskovits, 1991). In the case of these algorithms, the order implies that any variable in the list can only be influenced by variables that precede it to ensure there will not be any cycles in the learned graph.

The process of requiring a knowledgeable expert as input to an algorithm for learning BN structure from data has been criticised in the past as being counterproductive to the goal of learning BNs solely from data (Korb and Nicholson, 2011, p259). However, these criticisms are levelled at the dependence of expert knowledge for algorithms that are supposed to operate purely on data. In SEBN, the goal is to have all information about the structure of the BN elicited by experts anyway, so this is less of an issue. It is still a minor issue though, because the knowledge engineer will have to converse with an expert, rather than do everything strictly via surveys. The future work chapter discusses the prospect of moving this process into a survey as well (Section 8.7.4, p216).

4.4. Allocating a Subset of Questions to Each Survey Participant

Another way in which the problem of too many questions can be addressed is by only allocating a subset of questions to each expert. Thus, instead of each expert answering enough questions to produce an entire BN structure by themselves, they are producing fragments of BNs which are collated into an integrated BN structure. Figure 4.8 highlights the portion of the flowchart related to this process. When an expert begins the survey, they will be presented a set of questions that is only a subset of all possible questions.

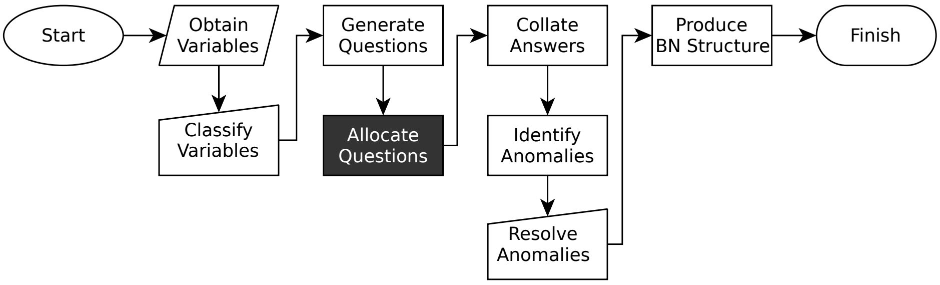
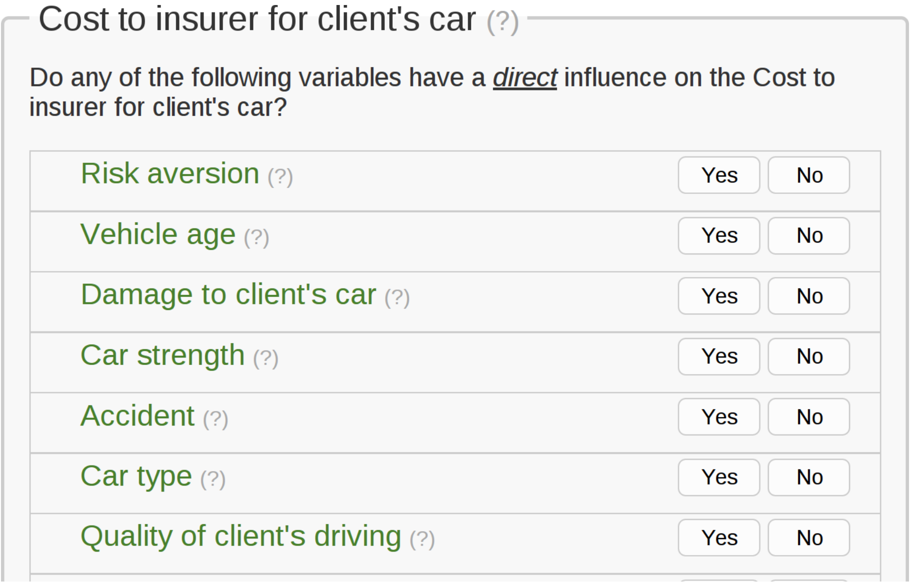


Figure 4.8.: After generating all of the survey questions, a subset of these are allocated to each expert so that no one person needs to answer every question.



Cost to insurer for client's car (?)

Do any of the following variables have a *direct* influence on the Cost to insurer for client's car?

Risk aversion (?)	Yes	No
Vehicle age (?)	Yes	No
Damage to client's car (?)	Yes	No
Car strength (?)	Yes	No
Accident (?)	Yes	No
Car type (?)	Yes	No
Quality of client's driving (?)	Yes	No

Figure 4.9.: Screenshot from BNE software configured with variables from the insurance network.

4.4.1. What is Allocated?

As discussed in Section 4.2, questions in the survey are of the form "Does X influence Y?". Although any individual question can be allocated to any expert, the BNE software allocates specific batches of questions of the form "Do any of $\{A, B, C\}$ influence Y?". That is, instead of asking "Does *risk aversion* influence *cost to insurer for clients car*?" and "Does *vehicle age* influence *cost to insurer for clients car*", they are grouped together to form "Do any of the following influence *cost to insurer for clients car*?" (Figure 4.9). These questions allow participants to focus their cognitive energy on one variable at a time, rather than jumping around to disparate questions. Section 8.7.5 (p216) discusses other ways in which questions can be grouped, for example, to reduce the number of parents of any given node making CPT elicitation easier.

4.4.2. Question Allocation Strategies

When a new expert starts to undertake a survey, they will be allocated a set of questions. The most basic allocation strategy is to allocate random questions to each expert. Alternatively, after classifying variables into specific categories (Section 4.3), then experts could nominate or be assigned to an area of expertise that they are most familiar with, resulting in only being allocated questions from variables in that area. This may not be suitable for the basic classes described at the beginning of Section 4.3 (Background, Mediating, Problem and Symptom variables), however could become very useful if using more specialised variable classes, as described later in that section. Note that this must be done with care, as Section 2.3.5 (p47) explains pitfalls with previous research that has neglected to properly define what constitutes an "expert" for the purposes of elicitation. The decision of why specific groups were allocated to specific experts should be communicated clearly to stakeholders.

If the survey was to be conducted in an offline setting (i.e. with pen and paper) then the questions would need to be allocated to participants before printing and sending mailing the surveys. In such a case, the level of non-responses to the survey would likely result in certain questions getting several participating experts to answer them, with others getting few or none.

Given the workflow presented in this thesis is for an online survey, the questions can instead be allocated at the time a participant signs up. This enables questions that have been answered the least number of times to be allocated to the next participant who logs in. The evaluation in Chapter 5 used this method, although it resulted in a slight problem. The BNE software chose questions for allocation if they had been *allocated to* the fewest people, whereas they should have been allocated after being *answered by* the fewest people. This issue was only identified after the evaluation surveys had been completed.

4.4.3. How Many Allocated Questions is Enough?

There are four measures which are important when deciding how many questions to allocate to each expert:

1. How many questions are there to be answered in total (n_{tot})?
2. How often should each question be answered by an expert (a)?

3. How many experts are available (n_e)?
4. How many questions are allocated to each expert (n_q)?

If it is unknown how many experts are available and able to contribute, then it is difficult to know how many questions each expert should be asked to answer. As such, often only n_{tot} is known for sure. This is shown in the following formula, restated in three ways:

$$n_e = \frac{n_{tot}}{n_q} \times a \tag{4.1}$$

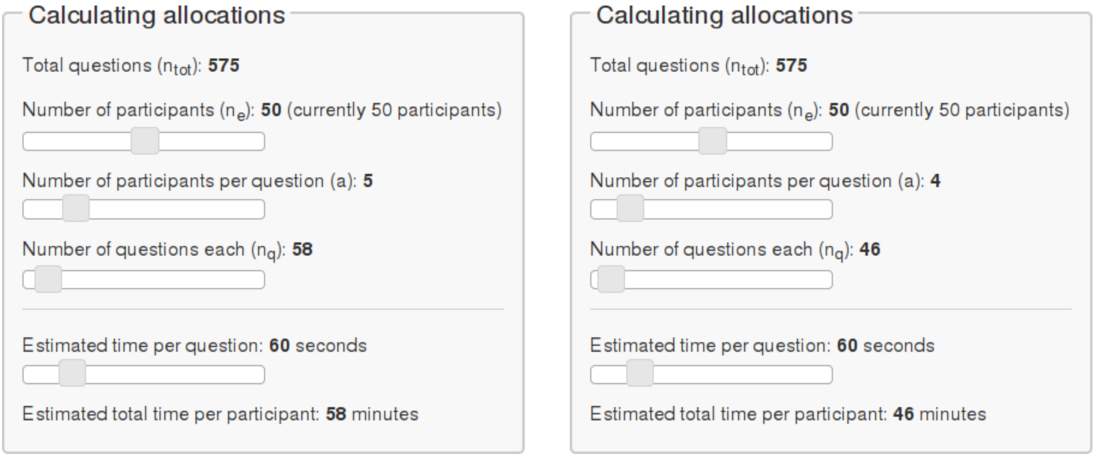
$$a = \frac{n_e \times n_q}{n_{tot}} \tag{4.2}$$

$$n_q = \frac{n_{tot}}{n_e} \times a \tag{4.3}$$

If there is a fixed number of experts that are known and have agreed to participate before the survey begins, then the equation becomes simpler. But this may not always be the case, in which case n_{tot} is the only fixed value, being defined by how many variables there are and what classes they fall into, once the variables and classes have been decided on (see Section 4.3). With these formulas, the required number of experts can be decreased by either: decreasing the required number of times a question should be answered (a), or increase the number of questions to allocate to each expert n_q, and hence increase the time required of them. Other than those three options, the only option remaining is to recruit more experts to participate (n_e). In the future work chapter, a method of active learning is discussed to allocate more hotly debated questions to more experts (Section 8.7.7, p217).

The BNE software provides a user interface for estimating the workload based on n_{tot}, n_e, a, and n_q. This enables administrators the ability to explore how many responses can be expected for their specific survey (Figure 4.10).

Advantages of Allocating a Subset of Questions The advantage of allocating only a subset of questions to each experts is that it helps to alleviate the knowledge



Calculating allocations

Total questions (n_{tot}): **575**

Number of participants (n_e): **50** (currently 50 participants)

Number of participants per question (a): **5**

Number of questions each (n_q): **58**

Estimated time per question: **60** seconds

Estimated total time per participant: **58** minutes

Calculating allocations

Total questions (n_{tot}): **575**

Number of participants (n_e): **50** (currently 50 participants)

Number of participants per question (a): **4**

Number of questions each (n_q): **46**

Estimated time per question: **60** seconds

Estimated total time per participant: **46** minutes

(a) When there are 575 questions and 50 participants, then requiring 5 experts per question results in 58 questions allocated to each expert, for an estimated time of 58 minutes.

(b) Lowering the value for a to 4 results in an estimated 12 minute time saving for each expert.

Figure 4.10.: Visually exploring parameters for allocating questions to experts using the BNE software.

bottleneck of KA. Doing so will help encourage contribution from a wider amount of experts, because they need only commit a smaller amount of time than they otherwise would have had to. Also, it provides for the option to ensure experts answer questions about their area of expertise, and are not forced into other areas that they are less familiar with.

Disadvantages to Allocating a Subset of Questions In traditional KEBN, each expert usually has the opportunity to contribute to the entire model as a whole. This is not the case when only a subset of questions are allocated to each expert, answered by participants on their own, without necessarily discussing with other experts, and then collated automatically without their further input (see Section 4.5). In addition to perhaps making the experts unsatisfied because they are not able to contribute to the entire model as a whole, there is also the possibility of anomalies being introduced into the BN structure once each of the experts' responses are brought together. The following section discusses in much greater detail the potential for these anomalies to occur, and also solutions for resolving them.

4.5. Collating Answers

The previous sections discussed allocating multiple questions to each expert, with each question being answered by multiple experts. After collecting the responses from each expert, there needs to be a principled way to decide what the “ground truth” of the question is. The goal of this part of SEBN is to produce a single BN structure. For each question “Does X influence Y?” and the multiple responses to that question, a final answer must be decided upon. Once this has been calculated, then the BN structure can be created. If the answer to “Does X influence Y?” is yes, then an arc exists from $X \rightarrow Y$, otherwise no arc is included in the structure. The choice to include or exclude an arc should take into account all responses to the relevant survey question (Figure 4.11).

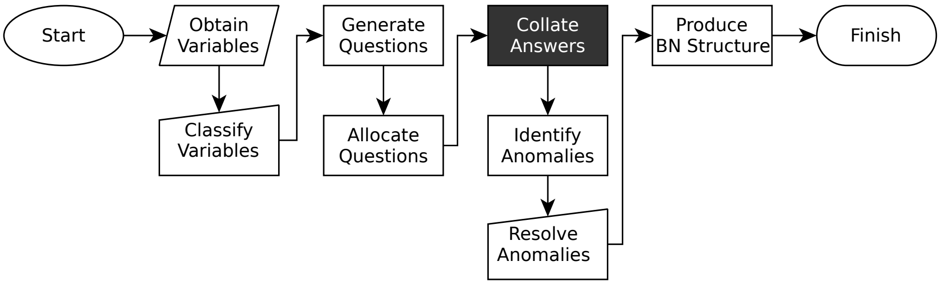


Figure 4.11.: Collation of answers is the part of the process where multiple answers to the same question, each from different experts, are combined to decide on the “best” answer.

This section will start with the majority vote algorithm which is simple but often dependable (Bachrach et al., 2012). This will be followed by a discussion of the Expectation Maximization (EM) algorithm (Section 2.4.2, p51). More advanced topics are discussed briefly earlier in this thesis (Section 2.4.2, p50) and could equally be used for collating responses, but are omitted due to the scope of this research project. The concluding chapter discusses how future work could extend SEBN to include a greater range of collation algorithms (Section 8.7.6, p217). The evaluation in Chapter 5 will further investigate the ability of majority vote and EM to collate participants responses together so that the trade offs as to which is preferable are better understood. The BNE software implements both majority vote and EM collation algorithms.

Majority Vote The majority vote collates responses to estimate the correct answer to a given question. Therefore, for this project, it entails calculating how the majority of experts answered a question of the form “Does X influence Y?”. If the majority say it does, then an arc between $X \rightarrow Y$ is inserted into the collated BN, otherwise the arc is omitted.

Note that it is not necessary to specify that 50% of respondents is the cutoff point for deciding on the “majority” response for a question. It may be that 20% of experts thinking a particular relationship exists is enough to be confident it should be included in the model. Chapter 5 evaluates different thresholds for what constitutes a majority and what effect that has on the resulting network structure.

Expectation Maximization Algorithm Dawid and Skene (1979) presents an algorithm for applying the EM algorithm to the task of combining multiple responses into a single model. To adapt their model for the purpose of collating responses into a BN, some terminology should be modified. In the Dawid and Skene (1979) model, the goal was to figure out the true state of a patient, based on the multiple responses they gave to different clinicians asking about their welfare. In BN collation, the goal is to combine multiple responses to a single question, in order to figure out what the best estimate at the correct answer to that question is. Therefore, the clinicians should be thought of as the experts, and the patients should be thought of as an arc in the collated BN. A response that a patient gives to a clinician is akin to a response given by an expert to a specific question of the form “Does X influence Y?”. Other than that, the algorithm can be adapted as is to the task of collating responses. The EM algorithm can also be implemented such that the prior probability of any given arc existing in a network is provided. As this prior probability increases, the collated BN structures will trend towards more arcs being included. Setting a lower prior implies that it is more difficult for an arc to be considered part of the collated structure, resulting in less arcs. Chapter 5 investigates priors of various levels to see how they impact the resulting network structure.

4.5.1. Note on Times When Multiple Models can be Created and Tested

It is worth noting that if SEBN was only used to elicit BN structures, and the CPTs were to be generated using data, then SEBN need not worry about one "final structure". Rather, the approach could be more akin to data driven, search and score based BN structure induction (Section 2.2.3, p29). Where previous structure induction techniques often make use of independence tests between variables to decide if they should be considered related, this approach could use the collated responses from the experts. This would result in many different candidate structures, which could be compared and searched through using relevant heuristics.

However, when the CPTs are going to be elicited from experts, it is preferable to do so only once. Thus, it is ideal to settle on one authoritative network structure before proceeding to the CPT stage.

When collating, there are some caveats which prevent a valid network from being able to be produced. The following section will discuss in greater detail certain considerations that will help ensure the elicited network is valid, and appropriate.

4.6. Identifying and Dealing With Anomalies

One issue with collating multiple responses into a single BN structure is the high potential emergence of anomalous structures. There are also optimisations that can be proposed based on the structures present in the network. Figure 4.12 highlights two processes for both identifying and resolve anomalies respectively. This section will discuss two different areas whereby anomalies can be identified and dealt with. This is followed by a discussion about identifying opportunities for further refining the resulting BN structure collated from survey responses (Section 4.6.3, p97).

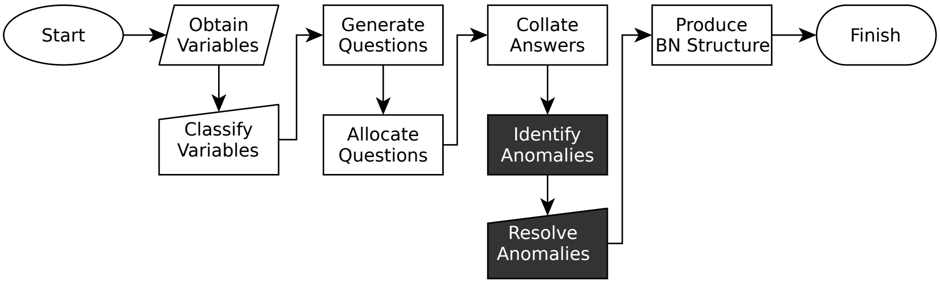


Figure 4.12.: After collating answers, the structure is searched for anomalies to be resolved by the experts.

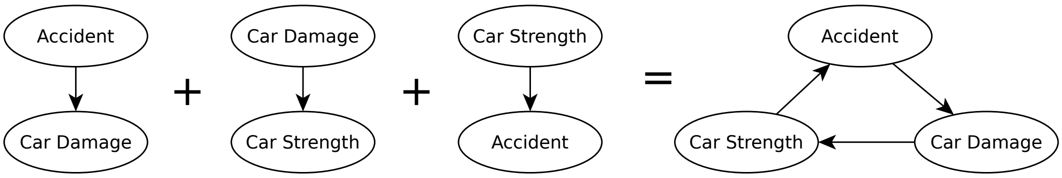


Figure 4.13.: Three seemingly valid causal relationships that form a cycle when combined . In order to make a valid BN structure, one of the valid assumptions will have to be disregarded to break the cycle.

4.6.1. Cycles in Network, Resulting in an Undirected or Cyclic Graph

Given the nature of directed graphs, it does not take much to introduce cycles, which prevent a Directed Acyclic Graph (DAG), one of the main requirements for a BN, and thus are not legal structures. For example, Figure 4.13 shows an example where two experts specify different and seemingly valid answers about the causal relationships between three variables. However, when combined, the network contains a cycle. This is one example, but cyclic relationships are not the only consideration.

In addition to the ease with which cycles can be introduced, the number of introduced cycles can be quite high, making it difficult to know the best arcs to remove to ensure the graph is acyclic. Figure 4.14 shows an example whereby a perfectly valid DAG is turned into a graph with 5 different cycles after the introduction of a single arc $F \rightarrow A$.

Cycles in the network structure are perhaps the easiest anomaly to identify, due

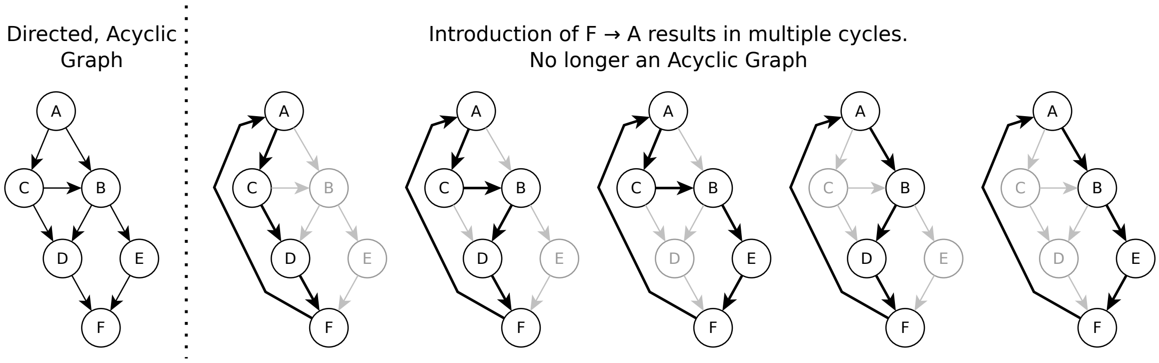

Directed, Acyclic Graph
Introduction of F → A results in multiple cycles. No longer an Acyclic Graph
A
B
C
D
E
F

Figure 4.14.: Example of how a legal DAG can end up with multiple cycles after introducing one relationship.

to the fact they don't need human input to be discovered. There are numerous algorithms to identify cycles (e.g. Bang-Jensen and Gutin, 2009; Johnson, 1975; Tarjan, 1973; Tiernan, 1970). There is also a solution to the presence of cycles - reverse or delete one or more arcs. The problem is prioritising which arcs should to be modified in order to provide a valid network structure, while having the smallest impact on the originally elicited structure. It should be noted that all arcs should be considered important, or else they would not have passed the collation stage.

Cycles can occur before collating questions too, based on the responses of a single expert. However it is not beneficial to resolve them at this stage. Doing so would require more effort from the experts, for little reward - given that the number of global cycles introduced in the collation stage would far outweigh the locally introduced cycles.

In the paper explaining the Grow-Shrink Markov Blanket Algorithm for inducing BN structure (Margaritis and Thrun, 1999), an algorithm is proposed which ensures the structure is acyclic. A graph which contains cycles often contains multiple cycles (e.g. Figure 4.14), and the larger the graph, the larger the potential for many cycles. Each cycle contains multiple arcs of the form $X \rightarrow Y$, some of which will appear in multiple cycles (but not multiple times in a single cycle). The algorithm from Margaritis and Thrun (1999) proposes reversing or removing the arc which is involved in the greatest number of cycles, and then restarting the process. In Figure 4.14, this would be the arch $F \rightarrow A$, involved in all 5 cycles, compared to, e.g., $A \rightarrow C$ which is only included in 3. Algorithm 4.3 paraphrases that algorithm in pseudo code.

Algorithm 4.3 Removing cycles from a directed graph, adapted from Step 4 & 5 of Margaritis and Thrun (1999, p4-5)

```
function RemoveArcs_MargaritisThrun($graph$)
    while HasCycles($graph$) do
        $\vec{cycles} \leftarrow$ FindCycles($graph$)
        $problemArc \leftarrow$ MostFrequentArc($\vec{cycles}$)
        $graph_{new} \leftarrow$ ReverseArc($graph$, $problemArc$)
        if HasCycles($graph_{new}$) then
            $graph_{new} \leftarrow$ RemoveArc($graph$, $problemArc$)
        end if
        $graph \leftarrow graph_{new}$
    end while
end function
```

Other Algorithms More Specific to BNs The above approach is generic in that it works with any DAG structure. The future work chapter of this thesis explores some different approaches which take into account specific features of BNs, such as sensitivity and arc strength (Section 8.7.8, p218).

4.6.2. Indirect vs Direct Relationships

Sometimes there is two separate causal mechanisms which allow one variable to influence another. If one of these mechanisms is via one or more mediating variables (i.e. X influences Z, because X changes Y which also influences Z) and the other influences the variable directly, then the result is an *indirect* and a *direct* relationship respectively. In the example shown in Figure 4.15, *Risk Aversion* → *Driving History* is the direct relationship, whereas *Risk Aversion* → *Senior Training* → *Driving Skill*→ *Driving History* is the indirect relationship that may explain the same phenomenon.

Although it is quite plausible to have a BN with both such relationships in them, it is also a hint that the entire indirect + direct relationship could be better encoded purely with the indirect relationship (van der Gaag and Helsper, 2002). These mediating variables enable BNs to properly encode the conditional independencies. It will also reduce the complexity of the BN by requiring a significantly smaller CPT for the child node, as it has less parents.

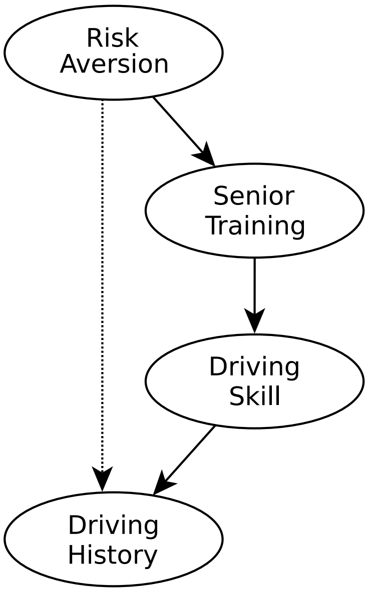


Figure 4.15.: Example of a *direct*, *potentially redundant* relationship in the insurance network (Binder et al., 1997) *"Risk Aversion → Driving History"*. It is potentially redundant because the *indirect* mediating chain of *"Risk Aversion → Senior Training → Driving Skill→ Driving History"* may be a better way to explain the relationship.

Due to the nature of this preference for indirect relationships, direct relationships that are better represented by indirect relationships will be referred to as *redundant*. Also, seeing as not all of these indirect and direct relationships result in a redundant relationship, they will further be termed *potentially redundant* until shown to be otherwise. The indirect relationship which hinted at the potentially redundant relationship will be referred to as the *mediating chain*.

This can be seen as similar to the Grow/Shrink algorithm (Margaritis and Thrun, 1999), in which a Markov blanket is built for each variable by iteratively adding variables that the variable is dependent on (the grow stage). However this includes variables which may be made independent by the inclusion of subsequent variables into the blanket. As such, each variable in the blanket is then iterated over in order to see if it can be removed (the shrink stage). The similarity with potentially redundant relationships is that these potentially redundant relationships may be quite important on their own. However, with the inclusion of more dependencies (ala the grow stage) mediating chains may be introduced. Once a mediating chain is identified and a direct relationship made redundant, then it should be removed (ala the shrink stage).

Potentially redundant relationships and their associated mediating chains can be identified using the algorithm shown in Algorithm 4.4.

Algorithm 4.4 Search for all the *potentially redundant* relationships in a DAG.

```
$\mathbf{R} \leftarrow \emptyset$        ▷ Potentially redundant relationships and their mediating chains
$\mathbf{G} \leftarrow$ DAG from Bayesian Network
$\mathbf{V} \leftarrow$ variables in $\mathbf{G}$
for all $\mathbf{V}$ as $v$ do
    $\mathbf{P} \leftarrow$ parents of $v$ in $\mathbf{G}$
    for all $\mathbf{P}$ as $p$ do
        Traverse $\mathbf{G}$ backward starting at $v$ but skip direct parent $p$
        if Hit $p$ while traversing then
            $r \leftarrow (p \rightarrow v)$        ▷ Potentially redundant relationship
            $m \leftarrow$ Traversed path from $v$ to $p$ ▷ Mediating chain which makes $r$ potentially redundant
            $\mathbf{R} \leftarrow \mathbf{R} \cup \{(r, m)\}$
        end if
    end for
end for
```

4.6.3. Optimisations to Aid in Subsequent CPT Elicitation

In addition to anomalies which cause undesirable behaviour, it is also possible to search for patterns which hint at opportunities to optimise the network structure. Certain configurations of nodes (known as the local structure of a BN) hint at the ability to elicit CPTs in a much more robust and also optimised way. In the following paragraphs, three different opportunities for optimisation are discussed, namely NoisyOR, NoisyMAX, and Ranked Nodes. Additionally, a “mixed bag” will be discussed, where knowledge engineers and experts are able to look at the network and make judgments on whether it can be improved in any way. Note that the evaluation in Section 5.6 (p141) will only investigate NoisyOR and NoisyMAX.

NoisyOR One type of distribution that needn't be parameterised by a full table of conditional probabilities that was identified early on with regards to development of BN theory, was the NoisyOR node (Pearl, 1986; Henrion, 1987). These are common when, for example, there is a node modelling the presence of a symptom and it has several possible causes - each of which is sufficient to trigger the effect, independent of the other causes. If each of the parent nodes and the effect node are all boolean, then a NoisyOR node can reduce the parameters required for the CPT from 2^n to n (where n is the number of parents).

- Criteria for NoisyOR node:
  - Child is boolean.
  - Child has more than 1 parent.
  - All parents are boolean.
  - Activation of any parent is enough to activate child, independent of other parents.

NoisyMAX A NoisyMAX node (Pradhan et al., 1994; Henrion, 1987) is a generalisation of a NoisyOR node. The differences are that both the target and the parent nodes are all partially ordered. As with NoisyOR, the presence of any parent node being activated is enough to activate the child. When the child is an n-ary node, then if any one of the parents takes a value which results in the child having a particular value, then the child will take that particular value. As such, the child node will take the maximum value that it is able do, based on the individual state of each parent. This will reduce the size of the CPT from $s_{child} \times \prod_{i=1}^{n} s_i$ to $\sum_{i=1}^{n} (s_i \times s_{child})$, where s_{child} is the number of states the child variable takes, s_i is the number of states that the parent variable i takes, and n is the number of parents.

- Criteria for NoisyMAX node:
  - Child is partially ordered.

  - Child has more than 1 parent.
  - All parents are partially ordered.
  - Any parent can cause child to take a given state, independent of other parents.
  - Child takes highest value of all possible based on individual parent conditioning.

Ranked Nodes Although BNs have been shown capable of including continuous random variables, many people still choose to discretize them to make the solution more tractable and understandable. As such, it is common to see variables in BNs with states such as $\{Very\,Low, Low, Medium, High, Very\,High\}$. Such variables can be termed "Ranked Nodes". Fenton et al. (2007) discussed how to reduce the burden of CPT elicitation for BNs that leverages ranked nodes. The requirements are that the child node, and all of its parents are ranked nodes. In the evaluation, Fenton et al. (2007) discuss two evaluations that resulted in reductions in CPT parameters by 84% and 93% respectively. Note that this would be more difficult to integrate into SEBN compared to NoisyOR and NoisyMAX as a lot more input from knowledge engineers and experts are required to ascertain whether the Fenton et al. (2007) approach can be used.

- Criteria for Ranked Node
  - Child is partially ordered.
  - Child has more than 1 parent.
  - All parents are partially ordered.

Manual Analysis by Knowledge Engineer In addition to creating a modular implementation that facilitates additional anomaly or optimisation detection, there is also the possibility for a more general approach. That is, present the network structure to an experienced BN practitioner, such as the researcher running the survey, and let them select arbitrary parts of the structure, and propose arbitrary questions for experts to answer. For example, if they noticed that a particular part of the structure could be simplified by adding a new node, then they'd select that part of the network, enter some text explaining what they are proposing, and also enter a few possible responses that they'd like to elicit from the experts. This way, in addition to the automated detection of specific anomalies and optimisation possibilities, there is also the ability for an experienced person to have the final say as to any strange looking areas of the structure.

Moving forward, such cases could be generalised if they are observed to occur frequently. Such generalisations could then be built into future versions of software such as BNE to make the elicitation process less burdensome.

4.7. Producing BN Structure

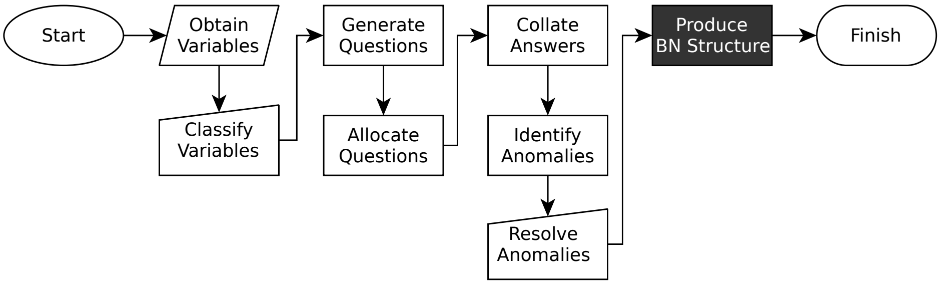


Figure 4.16.: After resolving anomalies introduced by collating multiple responses to each question, a final BN structure is output, ready for use in popular BN software packages.

Once arcs have been collated and any resulting anomalies resolved, then the BN structure is completed. The nature of the questions asked in the survey are that each question maps to an entry in a $n \times n$ adjacency matrix. Thus, it is a matter of taking the collated response to each question (Section 4.5) and creating an arc between the relevant variables if appropriate.

The BNE software supports outputting BNs to the Netica .net format (Norsys Software Corp, 2016), a summary matrix in HTML showing the strength of each arc (using the majority vote to indicate strength), and `.svg` graphics of the network structure using the graphviz software (Gansner and North, 2000), to visualise the network.

Thus, there is no requirement for manual intervention from the knowledge engineer to go from expert responses to a working model that can be imported into BN software such as Netica (Norsys Software Corp, 2016). This directly addresses Proposition 2 (p61), "Questionnaire method will require less time from researcher". Ensuring that they don't need to take extra steps to go from survey responses to working models which can be used straight away.

4.8. Chapter Summary

This chapter explained in detail the series of steps required in order to use surveys to elicit the *structure* of a BN. A summary shown in Figure 4.17 presents an overview of these steps. To evaluate this technique, Chapter 5 presents a detailed experimental evaluation of the technique presented in this chapter. The future work described in Section 8.7 (p214) discusses some ideas for extending and improving this technique. The following chapter presents an evaluation of SEBN as shown in this chapter. After this, Chapter 6 documents the remainder of SEBN in order to allow elicitation of CPTs to parameterise a BN structure. Combining both chapters results in a comprehensive method for eliciting entire, usable BNs from online surveys.

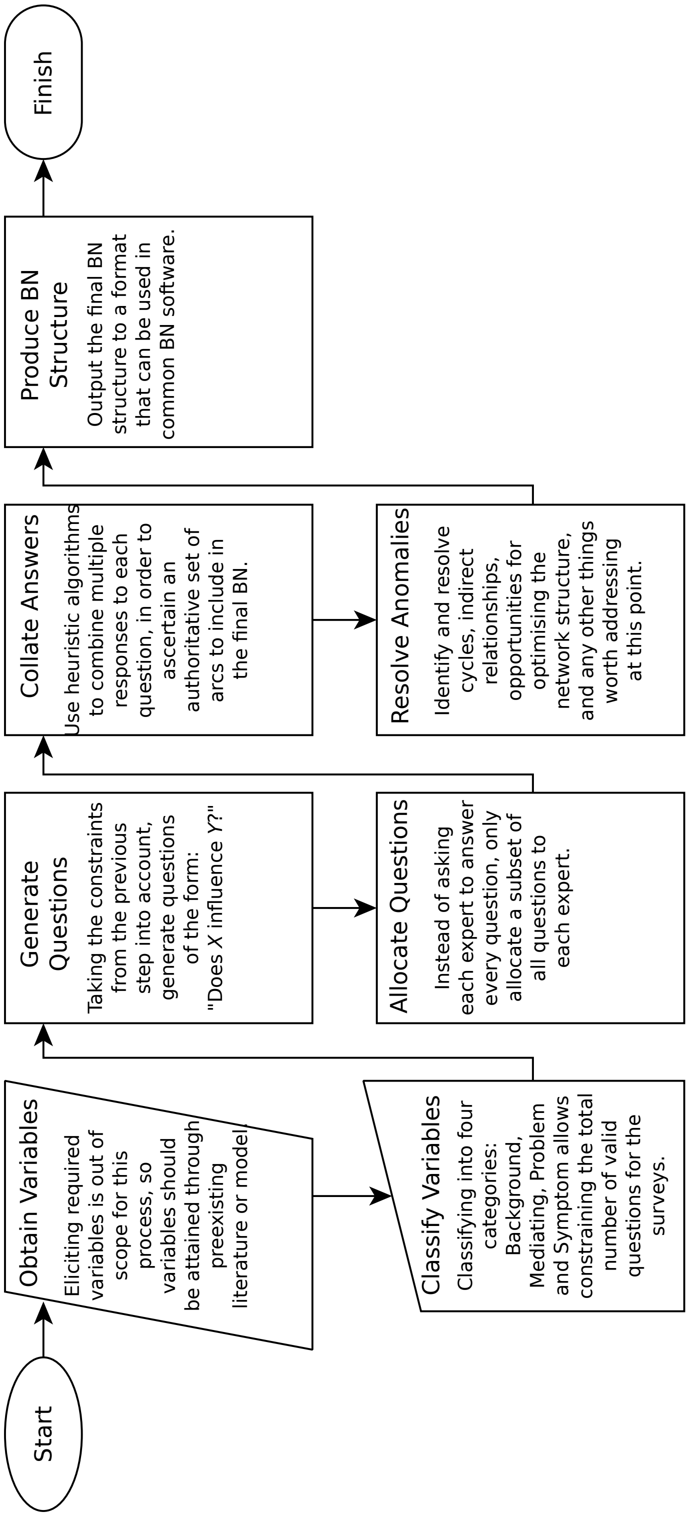

Start
Obtain Variables
Eliciting required variables is out of scope for this process, so variables should be attained through preexisting literature or model
Classify Variables
Classifying into four categories: Background, Mediating, Problem and Symptom allows constraining the total number of valid questions for the surveys.
Generate Questions
Taking the constraints from the previous step into account, generate questions of the form: "Does X influence Y?"
Allocate Questions
Instead of asking each expert to answer every question, only allocate a subset of all questions to each expert.
Collate Answers
Use heuristic algorithms to combine multiple responses to each question, in order to ascertain an authoritative set of arcs to include in the final BN.
Resolve Anomalies
Identify and resolve cycles, indirect relationships, opportunities for optimising the network structure, and any other things worth addressing at this point.
Produce BN Structure
Output the final BN structure to a format that can be used in common BN software.
Finish

Figure 4.17.: Summary of each of the stages in the process discussed in this chapter.

5. Evaluating Structure Elicitation

This chapter details the evaluation that took place for this project, with regards to eliciting the structure of BNs via SEBN (Chapter 4, p73). As discussed briefly in Chapter 3, there are two types of evaluation that took place. These are ex ante and ex post evaluations (Pries-Heje et al. (2008); Venable et al. (2012)), which roughly equate to "evaluation during the design process" and "evaluating the artifact after it is completed". These took place in both naturalistic and artificial settings.

This chapter will be organised as follows: Section 5.1 is dedicated to describing the method of evaluation. It discusses the experimental setup, the choice of participants, and briefly the online software system used to conduct the evaluation. Section 5.2 presents the results of collating the responses from the survey into several different candidate BN structures. This is followed in Section 5.3 by a comprehensive evaluation of the elicited BN structures. This section provides a critical analysis of the findings, comparing the network produced during evaluation to an existing gold standard. It presents several different metrics for which to compare the two networks. The time taken of participants is documented and discussed in Section 5.4, while Section 5.5 investigates whether SEBN is able to correctly infer the expertise of each participant based on their survey responses. Finally, Section 5.6 investigates two optimisations that can be performed to the BN structure, in order to ease subsequent CPT elicitation. The results from this chapter are used to address the propositions from Chapter 3 (p57) in the concluding chapter (Section 8.2, p195).

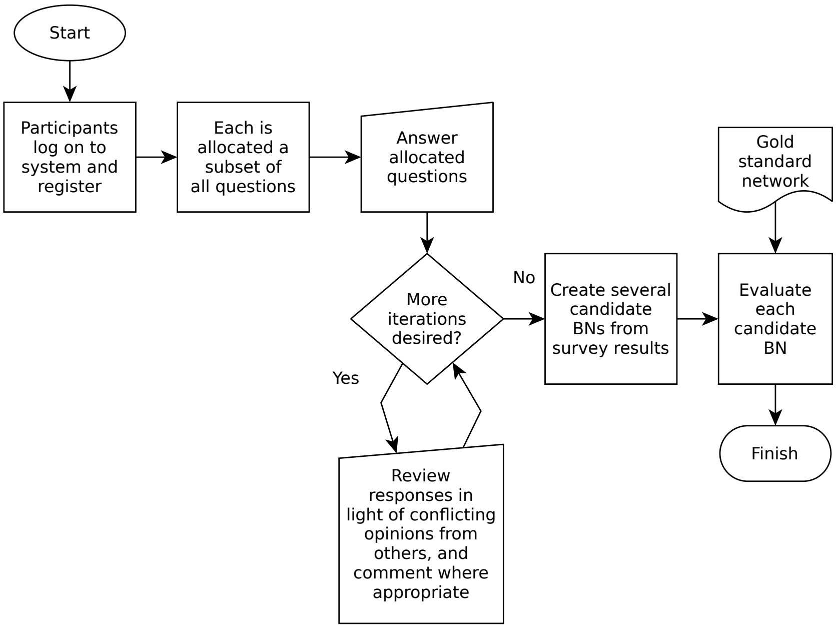

Start
Participants log on to system and register
Each is allocated a subset of all questions
Answer allocated questions
More iterations desired?
Yes
Review responses in light of conflicting opinions from others, and comment where appropriate
No
Create several candidate BNs from survey results
Gold standard network
Evaluate each candidate BN
Finish

Figure 5.1.: A broad overview of the process for conducting the evaluation survey and analysing the results.

5.1. Experimental Method Used for Evaluation

In order to evaluate the propositions described in Section 3.3 (p60) and addressed in Section 8.2, this evaluation conducted an online survey using SEBN via the BNE software. This section will discuss the evaluation process, survey participants, and the software used, and the duration of the survey. Discussion on the results of the evaluation will be held off until later in this chapter.

5.1.1. Overview of Experimental Method

This evaluation elicited a BN structure using the method described in Chapter 4 (p73) in order to compare with a known gold standard. Figure 5.1 shows a high level overview of the evaluation. The domain chosen for the evaluation was car insurance risk assessment due to the availability of an existing, published BN which can be used as a gold standard for which to compare the survey results to.

The evaluation took place using an iterative approach. When answering questions, participants were asked to provide comments on why they thought there was a causal mechanism between two variables. Once the first set of questions were answered by all participants, the software was configured for the next iteration. In this and subsequent iterations, participants were invited to come back and review their answers in light of decisions and comments made by other participants. This is highlighted in Figure 5.1, whereby responses are continually reviewed each iteration.

5.1.2. Note on the Iterative Nature of the Evaluation Survey

Although SEBN as proposed in this thesis does not suggest an iterative approach to surveying experts, it was still included in this analysis for the sake of investigation. It was hoped that asking participants to consider their responses in light of comments made by others would help facilitate discussion and aid them in coming to the optimal solution. However, as shown in Section 5.1.4, the participation drop-off rate between iterations was sufficiently problematic that there was not enough responses in the later iterations.

5.1.3. Choice of Existing "Gold Standard" BN

The evaluation was conducted with respect to an existing network in the field of risk assessment for car insurers, which will henceforth be referred to as the "insurance network" (Binder et al., 1997). The structure of the network was elicited by researchers and verified by an expert, similar to what might be expected in traditional KEBN[1]. It was subsequently parameterised using historical data. As such, the parameterisation was not a traditional KEBN approach in the way it made use of experts. The BN consists of 27 variables, 52 arcs, and a total of 1419 conditional probability values.

[1] The original insurance network article (Binder et al., 1997) doesn't clarify how the structure was elicited. However subsequent private communication with the authors explained that the network was elicited by the researchers based on their knowledge of the risk factors, and then shown to somebody with experience in the insurance industry.

The insurance network has been published, peer-reviewed, and is publicly available, including the entire network and its probabilities[2]. Also, it is from a field that is not too specialised, in order to facilitate the recruitment of participants. This is elaborated on in a following section, which documents the process of recruiting participants for the evaluation survey.

5.1.4. Recruiting Participants for Survey

This section begins with a discussion of the rationale behind the participants that were chosen. This is followed by an analysis of the participation rate among those who did sign up for the evaluation survey.

Choice of Participants

The participants for the evaluation study were drawn from a pool of lay people who have experience driving. This is somewhat similar to how psychology researchers often utilise undergraduate students as a proxy for other populations (Wintre et al., 2001). Although the practice is undesirable, given that undergraduates are not representative of the broader adult population, it is still very common. Wintre et al. (2001) explain that if the practice is to continue, research should at least make a concerted effort to discuss the limitation of their findings in terms of being generalisable to broader populations.

In this study, the choice to use lay people instead of experts came with pros and cons. The pros are that they are easily accessible, numerous, and allowed for multiple evaluation surveys to be conducted at little cost. The cons are that they are not as expert as traditional experts, have less vested interest in trying to answer the survey questions correctly, and have less of an appreciation for what they know and don't know. Despite this, they are a suitable population of participants in this context for the following reasons:

> *By definition, people with experience driving have some level of expertise.*

[2] Not all published networks include all of the probabilities and relevant parameters.

Although they are not what one would term “experts” in the field of car insurance, they do have expertise with driving, and hence useful knowledge about the relationship between variables involved with driving. This information can still be elicited and incorporated into a BN model.

> *Variables in the car insurance BN are concepts that drivers would be familiar with.*

Examples of variables used in the car insurance network are the cost of cars, driving skill, safety features of vehicles, etc. For the most part, these variables and the relationships between them represent concepts that lay people with experience driving would understand. There is almost no insurance specific variables or jargon in the gold standard BN.

> *The survey utilised theories inspired by the field of crowd sourcing.*

Despite SEBN as described in Chapter 4 and Chapter 6 being targeted towards *experts,* it is convenient that the collation techniques used by the method are drawn from crowd sourcing. The reason is that in crowd sourcing, the goal is to elicit information from multiple people of unknown expertise, and decide which information is useful and which is not. If there are adversarial participants (intentionally answering incorrectly) or people of less expertise, their answers will be given less weight.

> *If the survey produces good BN with people of less expertise, it should produce a better network with those of higher expertise.*

Finally, and most importantly, there was no reason to believe that people with less expertise would contribute *better* knowledge than proven, qualified experts. Thus, if the evaluation was able to show that people of less expertise can produce a sufficiently good BN, then it is reasonable to assume that qualified experts would produce a *better* BN. Unfortunately, not much can be said if the reverse was to occur. That is, no information is gleamed about what experts may be capable of if lay people are unsuccessful. This is a limitation of this research and discussed further in Section 8.7.1 (p214).

The choice of participants is evaluated in a post-hoc manner in Section 5.7 to see what can be learnt and how to improve future evaluations of SEBN.

Summary of Participation

Chapter 4 discussed specifics about the choice of how many experts to recruit, and how many times each question needs to be answered by a different expert (p87). This evaluation left the number of experts (n_e) as variable, in the hope that as many people as possible could sign up. Experience with an earlier ex ante evaluation showed that recruiting participants for the survey before it begins resulted in people initially agreeing to participate, but then being unable to respond once the survey started. Due to n_e being left variable, the number of questions asked of each expert (n_q) or the number of times each question was allocated to an expert (a) had to be decided upon. As individual participants registered with the online system to participate, they were allocated a number of questions, for which the goal was to have each participant have the same number of questions. Thus, n_q was fixed at 4 variables for which to ask "Does any of $\{A, B, C, ...\}$ influence X?". The configurations of variables and variable classes in this survey meant an allocation of 4 variables resulted in 88 specific "Does A influence X?" questions that required answering.

Figure 5.2 shows that even before the first question, 11 of 43 registered participants dropped out. The first iteration started with 32 of which 23 finished. For the remaining two phases 11 participants dropped out, resulting in 14 participants of the original 43 completing all three phases.

Figure 5.3 shows how often each of the 24 groups of questions of the form "Do any of $\{A, B, C, ...\}$ influence X?" were answered by the end of each iteration. Note how there is one variable *PropCost* which received zero answers in the second and third iteration. In addition, six variables only had a single response in the final iteration, which means that there is no benefit to collating responses from multiple participants, as there are not multiple participants to collate from. Despite the slowing drop off rate shown in Figure 5.2, and the fact that there were still 14 participants who completed the final stage, there was not enough data to collate responses after the third phase. As such, only the first round of data was used, and SEBN described in Chapter 4 does not include an iterative approach.

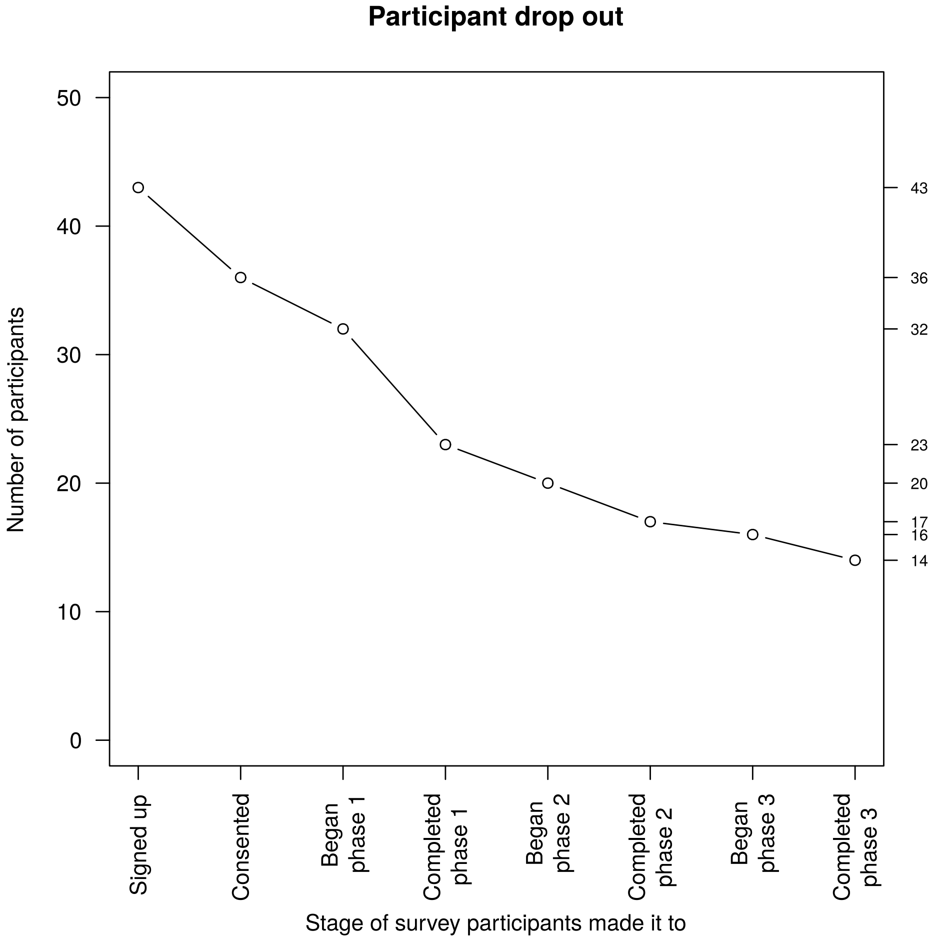


Figure 5.2.: Visualising the participation dropout between iterations.

5.1.5. Software Used for Evaluation Survey

To conduct the online elicitation survey, custom software was written which implemented SEBN as described in Chapter 4. This section will briefly introduce the software, and a more in depth discussion about decisions taken in the design and development of the software is presented in Appendix A. The software is called BNE (BN Elicitator, Serwylo, 2013) and is published under the GPLv3 open source license.

The software is a web application that enables participation from anybody with a

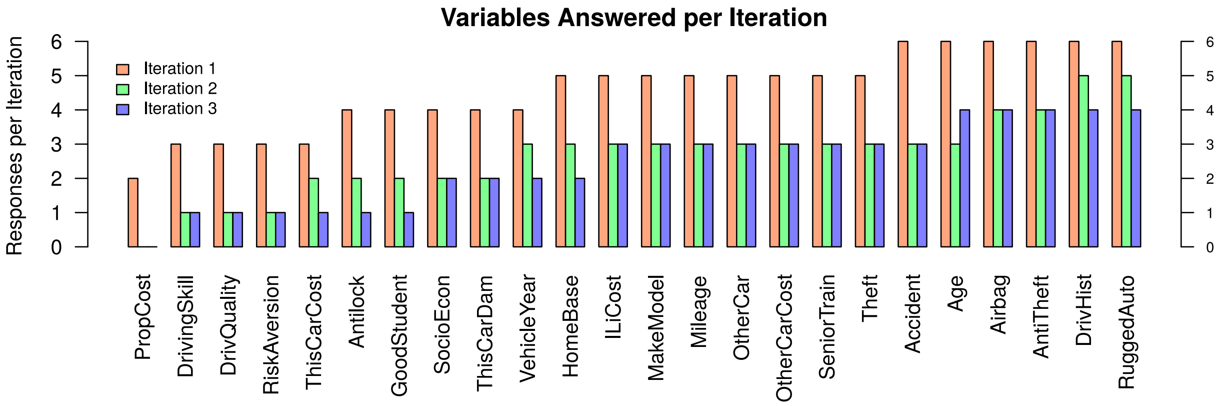


Figure 5.3.: Number of responses received about each variables questions for each iteration.

web browser connected to the internet. It facilitated people registering themselves (using either an email and password or a Facebook account), or the administrator selectively recruiting experts and granting them access to the survey. There is no limit to the number of people who are able to participate in a survey, and each new user will automatically get allocated a set of questions. The result of a larger number of people signing up means that each question gets answered by a greater number of people.

The administrator of the survey software has the freedom to decide when to move onto the next iteration. After the first iteration is closed, people are no longer allowed to register to participate in the survey, only those already registered can proceed to subsequent rounds. At this point, they could view the responses given by other participants. To reduce the burden on the participants, questions which had 100% agreement in the prior iteration are hidden in subsequent rounds .

5.1.6. Duration of Evaluation Survey

The evaluation took place over a period of approximately one month in 2013. Three iterations were performed that took 9, 11, and 14 days respectively. The initial goal was to have three iterations at one week each, but it was made flexible to ensure a higher response rate. Each participant was alerted to the fact that it may take up to a month for all iterations to be completed, at a time of about 30 minutes each. The time at which the iteration was moved to the next iteration was at a time chosen by the researcher, after the amount of responses slowed down sufficiently.

5.2. Constructing the BN Structure From Survey Responses

As discussed in Section 3.3.3 (p65), it is important for SEBN to be capable of producing a valid and appropriate BN. If the method is not able to produce a valid BN or it is only able to produce valid but poor BNs, then it cannot be recommended as a technique for creating BNs at this stage. As such, a lot of the evaluation will be focused on the output of the survey process, and whether it constitutes a "good" BN, for various definitions of the word "good".

Firstly, Section 5.2.1 will discuss how the various responses to questions were collated together into a directed graph. Then, Section 5.2.2 investigates the process of ensuring this graph is acyclic, a key requirement for using a graph as the structure of a BN. Once this has been done, the remainder of this section is devoted to visualising the resulting structures.

5.2.1. Collating Responses into a Directed Graph

As discussed in Section 4.5 (p90), responses can be collated together in a number of ways. This evaluation compares two different methods: majority vote and the EM algorithm, referred to as "Dawid & Skene" throughout this thesis as Dawid and Skene (1979) were the first to apply this statistical technique to collating multiple responses. Each of these two approaches are capable of producing multiple candidate networks, which are discussed in detail below.

Output from the Survey Software The result of the online surveys is a set of multiple participants' responses to questions of the form: "Does X directly influence Y?".

Overview of the Different BN Structures Being Evaluated Shown in Table 5.1 is a list of different networks which will be measured during this evaluation section. The Majority Vote (Maj) and Dawid & Skene (DS) network structures represent those which are the result of collating responses from the evaluation survey in varying ways. Both together are referred to as the *Survey* networks.

The result of the majority vote algorithm varies depending on the threshold of "how many people are required to agree" for an arc to be included in the final structure[3]. As the number of people required for agreement *decrease*, the number of arcs in the network will *increase*. This was done using custom software written for this project and included in the BNE software. The six Maj structures are collated by including all arcs which at least 2, 3, 4, 5 or 6 participants agreed on, resulting in five different network structures for evaluation[4]. They were created by counting the number of responses advocating for a particular arc, and if it was above the threshold, including it in the network structure.

DS structures are collated by running the EM algorithm over participants responses and specifying a prior probability of any particular relationship between two variables existing. With no other information other than the *Gold* network, the prior probability of an arc should be ~7%. This is because the *Gold* network has 52 out of a potential 702 arcs[5]. Thus, this chapter evaluates networks with priors of 0.1%, 1%, 5%, 10%, 15%, 20%, 25%, 30% and 35%. If the prior is *decreased*, then the number of arcs will also *decrease*. The reason for this choice of priors is because empirical findings during evaluation showed that $0 < prior \leq 0.1\%$ resulted in a network structure with zero arcs, the same as $Other_{zero}$ described below. Additionally, $prior > 35\%$ included so many arcs as to represent a network structure that was computationally intractable to analyse. The details of the algorithm are explained in greater detail in the original paper (Dawid and Skene, 1979) or in Section 4.5 (p90). The implementation of the EM algorithm incorporated into the BNE software was from the *Troia* software package (Project Troia, 2013).

[3] This means that technically the word *majority* is incorrect, but the term "majority vote" has enough meaning to make it both worthwhile and useful.

[4] The Maj_1 structure had so many cycles as to make the cycle detection algorithm incapable of calculating them all due to computational constraints. Thus, it was left out from the analysis.

[5] The 702 possible arcs come from $n^2 - n$, given that variables cannot influence themselves. It also does not take into account the variable classes discussed in Section 4.3 (p80)

Label	Description
Gold	“Gold Standard” Existing network from literature (Binder et al., 1997).
Maj_2	“Majority vote” Subscript represents the number of participants required to agree before arc is included in this BN.
Maj_3	
Maj_4	
Maj_5	
Maj_6	
Maj	$\{Maj_2, \ldots, Maj_6\}$
$DS_{0.001}$	“Dawid & Skene” EM algorithm for collation (Dawid and Skene, 1979). Subscript represents the prior probability of any single arc between two variables being included in this BN. Lower prior means less likely to include arcs, and thus smaller networks.
$DS_{0.01}$	
$DS_{0.05}$	
$DS_{0.10}$	
$DS_{0.15}$	
$DS_{0.20}$	
$DS_{0.25}$	
$DS_{0.30}$	
$DS_{0.35}$	
DS	$\{DS_{0.001}, \ldots, DS_{0.35}\}$
$Learnt_{mmhc}$	“Learnt from existing algorithms” These are learnt by sampling data from the *Gold* network then using that data as input to the $mmhc$, $rsmax2$, and $tabu$ algorithms.
$Learnt_{rsmax2}$	
$Learnt_{tabu}$	
$Learnt$	$\{Learnt_{mmhc}, Learnt_{rsmax2}, Learnt_{tabu}\}$
$Other_{rand}$	“Random network” Generated from `random.graph` function in the *bnlearn* software (Scutari, 2010).
$Other_{zero}$	“Zero connections” Each node is independent of all others.
$Other$	$\{Other_{rand}, Other_{zero}\}$
$Survey$	$Maj \cup DS$
$Eval$	$Survey \cup Learnt \cup Other$

Table 5.1.: Various different BN structures that will be referred to throughout this evaluation, and their meaning. The *Maj* and *DS* (aka *Survey*) structures are created from data obtained during evaluation, while the remainder are used in order to have something to compare these to.

The three *Learnt* network structures were learnt using existing algorithms provided by the *bnlearn* software (Scutari, 2010). A data set to use for learning the networks was sampled from the *Gold* network and then given to the relevant learning function.

Finally, the $Other_{rand}$ and $Other_{zero}$ networks are provided for comparison. $Other_{rand}$ is constructed using the `random.graph` function from *bnlearn*, whereas $Other_{zero}$ is a network with zero arcs.

5.2.2. Removing Cycles From the Collated Graph

Many of the evaluation mechanisms being used later in this chapter depend on a valid BN structure being present. Thus, it was important to go through the anomaly resolution phase before evaluating[6]. Section 4.6 (p92) discusses how to ensure that a valid DAG is output after the process of combining responses together. This section discusses in more detail how that was undertaken for this specific evaluation.

The implementation in the BNE software follows the Margaritis and Thrun (1999) algorithm described in Section 4.6 (p92). It made use of the *JGraphT* (Naveh and Contributors, 2015) implementation of the Johnson (1975) cycle detection algorithm.

Cycle removal for $DS_{0.05}$ Table 5.2 shows the arc reversals that occurred to ensure the $DS_{0.05}$ network structure was acyclic. Unfortunately, some of the reversals result in seemingly non-causal relationships. For example, the previously causal relationship of *Age*→*RiskAversion*, is reversed, resulting in *RiskAversion*→*Age*. Although a relationship between the two has been observed by participants taking the survey and thus the process should retain some sort of association, it is undesirable. Specifically, it may cause issues when subsequently eliciting CPT values, where non-causal questions are then posed. This could greatly confuse people and

[6] Note that some of the evaluation metrics, such as SHD, are able to be calculated without removing cycles. However, it was deemed unhelpful to compare invalid *Survey* network structures (i.e. those with cycles) to others which have stricter constraints, such as the *Learnt* structures.

result in worse CPT values and in general a BN. This is discussed in greater detail in the Future Work chapter (Section 8.7.8, p218).

Iteration	# Cycles	Worst arc	Appeared in (cycles)
1	34954	*ThisCarCost* → *Age*	22570
2	3355	*ThisCarCost*→ *MakeModel*	9498
3	2886	*DrivHist*→ *DrivQuality*	1443
4	1443	*DrivHist*→ *DrivingSkill*	933
5	510	*OtherCar*→ *DrivingSkill*	351
6	159	*SeniorTrain*→ *RiskAversion*	137
7	22	*DrivHist*→ *SeniorTrain*	5
8	17	*RiskAversion*→ *Age*	4
9	13	*Accident*→ *SeniorTrain*	3
10	10	*Theft*→ *HomeBase*	2
11	8	*DrivQuality*→ *DrivingSkill*	2
12	6	*Airbag*→ *MakeModel*	2
13	4	*Theft*→ *AntiTheft*	1
14	3	*Age*→ *DrivingSkill*	1
15	2	*Airbag*→ *VehicleYear*	1
16	1	*VehicleYear*→ *Antilock*	1

Table 5.2.: Iterations required to turn the $DS_{0.05}$ graph into a DAG using the Margaritis and Thrun (1999) algorithm. Each of the 16 iterations resulted in the arc being reversed, not removed (See Appendix B for details of other *Survey* structures).

Figure 5.4 shows an overview of each of the evaluation networks, and how many arcs required modification to become a DAG. Note how as the *Maj* threshold decreases or the *DS* prior increases, the number of arcs implicated in a cycle increase, but less so with the *DS* networks.

5.2.3. Summary Matrices

In order to get a broad overview of the *Gold* and *Survey* network structures, summary matrices have been employed (similar to Flores et al., 2011). These are adjacency matrices whereby boolean “does this arc exist in the network” values are shown by a black dot in this evaluation and continuous “strength of the rela-

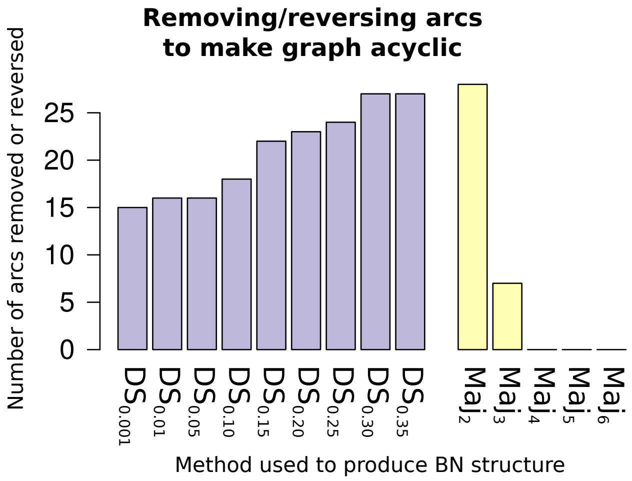
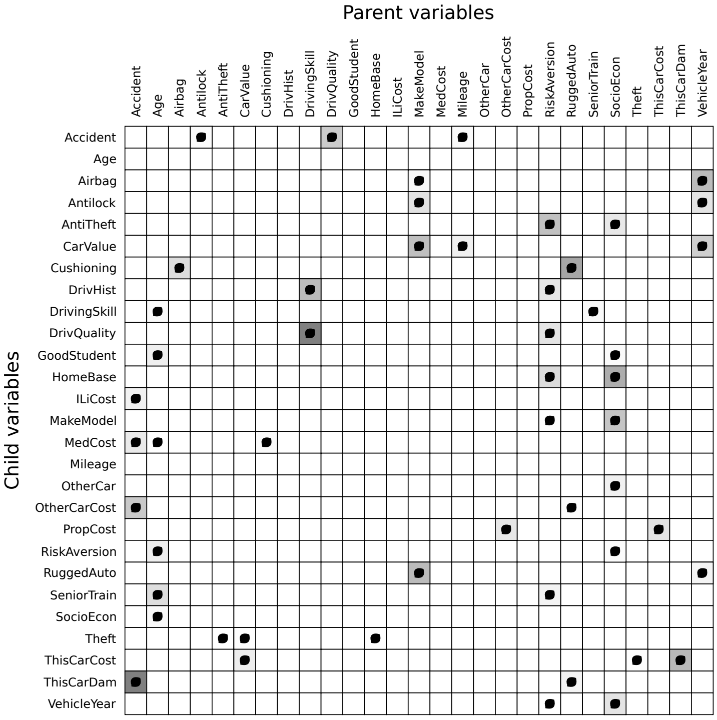


Figure 5.4.: Graph showing how many arcs were removed in order to get a DAG, for each *Eval* network.

tionship" values are shown by the darkness of the background shading (darker is stronger).

It is worth noting that the strength of a relationship in the *Gold* network is not comparable to the strength in the *Survey* networks, because each uses a different scoring algorithm to determine strength. However, they are useful in terms of intra-algorithm (e.g. Maj_3 vs Maj_4) and intra-network (e.g. $x \rightarrow y$ vs $x \rightarrow z$ in $DS_{0.1}$) strength comparisons.

Original Network To begin, Figure 5.5 shows a summary matrix of the original network. The strength of each arc was calculated by using the `arc.strength` function from the *bnlearn* software package (Scutari, 2010). This in turn used a technique whereby the overall BIC (Schwarz, 1978) was calculated for the network. Then, each arc was removed from the whole network and the score re-calculated. Those which caused the BIC to change greatly are said to have a strong relationship, whereas those which tend not to affect the resulting BIC are weaker. For example, in Figure 5.5 the relationship *SocioEcon* → *GoodStudent* is a particularly weak arc, whereas *Accident* → *ThisCarDam* is very strong.



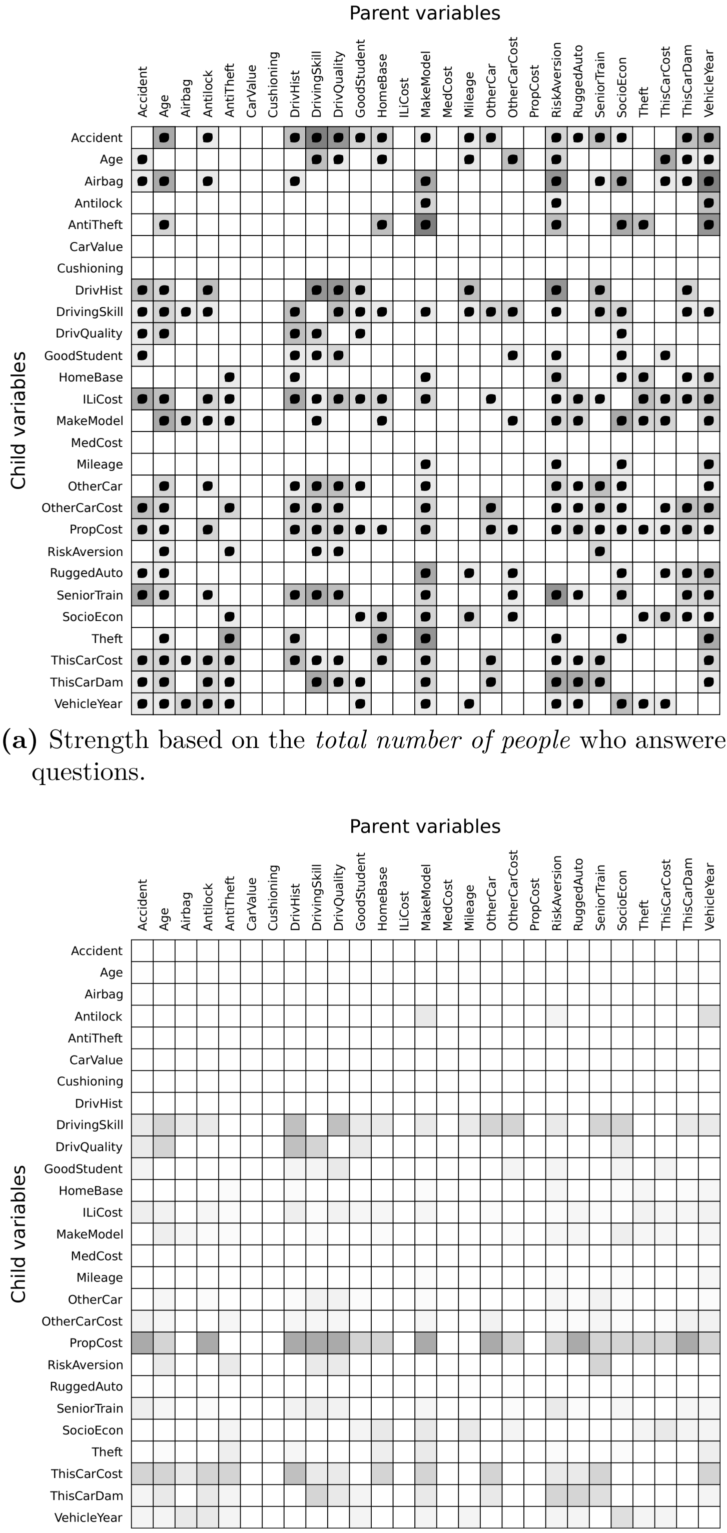
	Accident	Age	Airbag	Antilock	AntiTheft	CarValue	Cushioning	DrivHist	DrivingSkill	DrivQuality	GoodStudent	HomeBase	ILiCost	MakeModel	MedCost	Mileage	OtherCar	OtherCarCost	PropCost	RiskAversion	RuggedAuto	SeniorTrain	SocioEcon	Theft	ThisCarCost	ThisCarDam	VehicleYear
Accident				●						●						●											
Age																											
Airbag														●													●
Antilock														●													●
AntiTheft																				●			●				
CarValue														●		●											●
Cushioning			●																		●						
DrivHist									●											●							
DrivingSkill		●																				●					
DrivQuality									●											●							
GoodStudent		●																					●				
HomeBase																				●			●				
ILiCost	●																										
MakeModel																				●			●				
MedCost	●	●					●																				
Mileage																											
OtherCar																							●				
OtherCarCost	●																				●						
PropCost																		●							●		
RiskAversion		●																					●				
RuggedAuto														●													●
SeniorTrain		●																		●							
SocioEcon		●																									
Theft					●	●						●															
ThisCarCost						●																		●		●	
ThisCarDam	●																				●						
VehicleYear																				●			●				

Figure 5.5.: Summary matrix for the *Gold* network structure. Dark circles indicate the presence of an arc, and the darkness of the background colour indicates its strength.

Majority Vote Network The matrix in Figure 5.6a shows the summary matrix for the network which includes all arcs agreed upon by at least one participant[7]. The shades of grey indicate how many people out of six voted for a particular relationship. Each question was allocated to six participants, but not everybody answered all the questions that were allocated to them. As a result, Figure 5.6b shows the same matrix, but this time the weights are shaded according to the *percentage* of participants who included them. All of the variables that are said to influence *PropCost* are much stronger. This is because only two people answered questions relating to *PropCost* (see Figure 5.3), resulting in only two votes being needed for a "strong" relationship (100% consensus among respondents).

Finally, Figure 5.7a shows the summary matrix for the Maj_3 network. This was

[7] Note that with all summary matrices of *Eval* networks, there are variables which have been removed from the network, resulting in empty rows/columns in the matrix. See Appendix D (p257) for further details.



(a) Strength based on the *total number of people* who answered questions.

(b) Difference between strength of arcs in Maj_1 network when measured as the *percentage of people* who answered questions less the *absolute number of people* who responded.

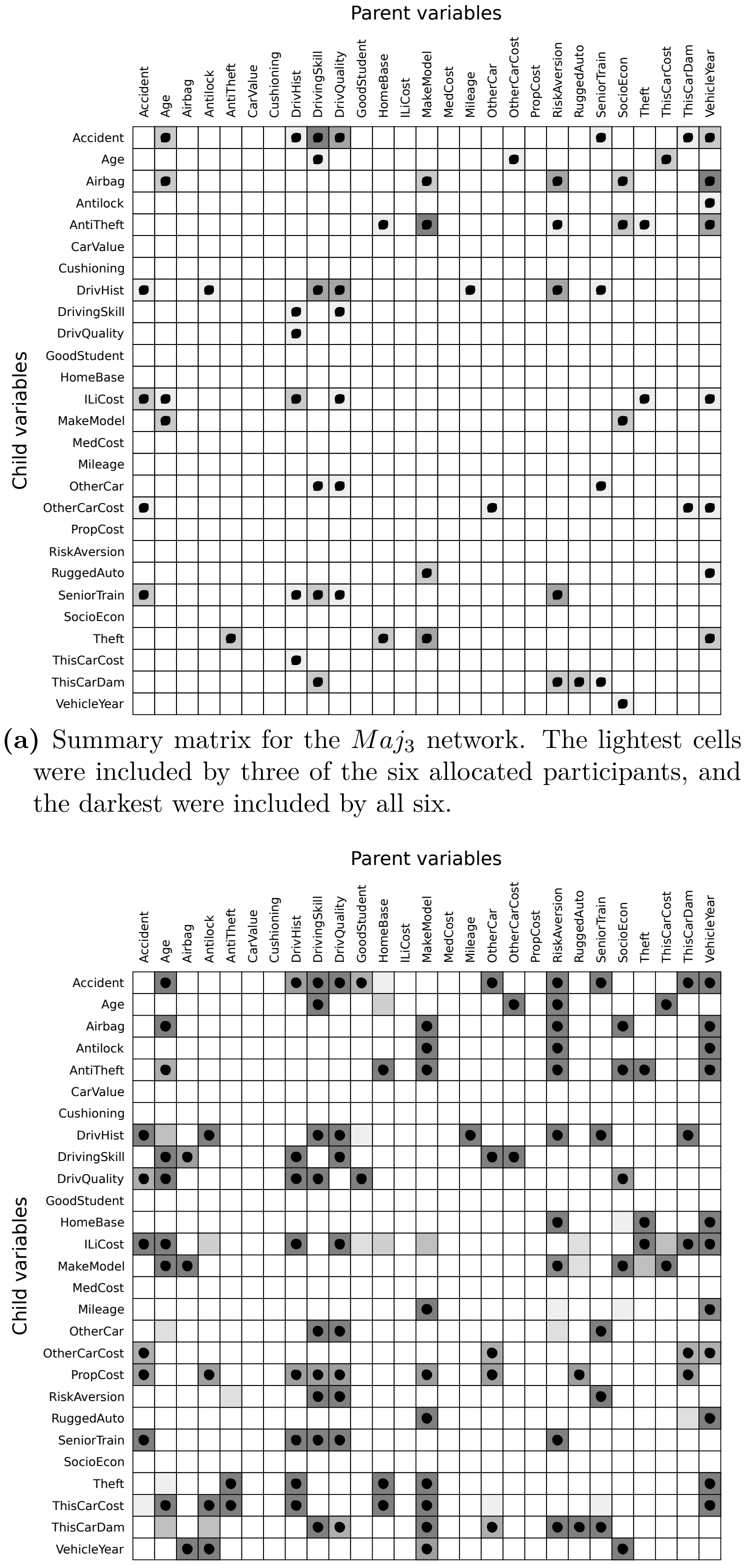
Figure 5.6.: Summary of the Maj_1 BN. Black dots indicate a relationship between the two variables, while the darkness of the cell indicates the strength of the relationship.

selected for inclusion here because later sections show that it is one of the “better” network structures.

Dawid & Skene Network The summary matrix in Figure 5.7b shows the strength of arcs in the DS network structures. It encodes the chance that the EM algorithm would include any given arc in the network structure. The strength is ascertained by considering the lowest prior probability at which an arc is still included. If an arc is included *despite* a very low prior, then it is considered stronger. If an arc is not included unless the prior is quite high, then it is assigned a weaker strength. The arcs in the $DS_{0.15}$ network are shown as black circles in the centre of a cell.Note how there is quite a large number of arcs compared to the *Gold* network and also the Maj_3 network.

Difference Matrices In order to compare the *Survey* networks to the *Gold* network to see how closely they align this section presents two *difference matrices* (Figure 5.8). These are similar to the summary matrices shown earlier except they show the intersection of two network structures. The *Gold* network is shown as squares whereas the *Survey* network is shown in black circles. Also included are arc strengths from the original network. This allows visual inspection of whether missing or retained arcs are important to the original network. For example, *VehicleYear* → *Airbag* in the top right of Figure 5.8a is a strong relationship which is included in both networks, whereas *Accident* → *ThisCarDam* is a strong relationship that is not present in either elicited network.

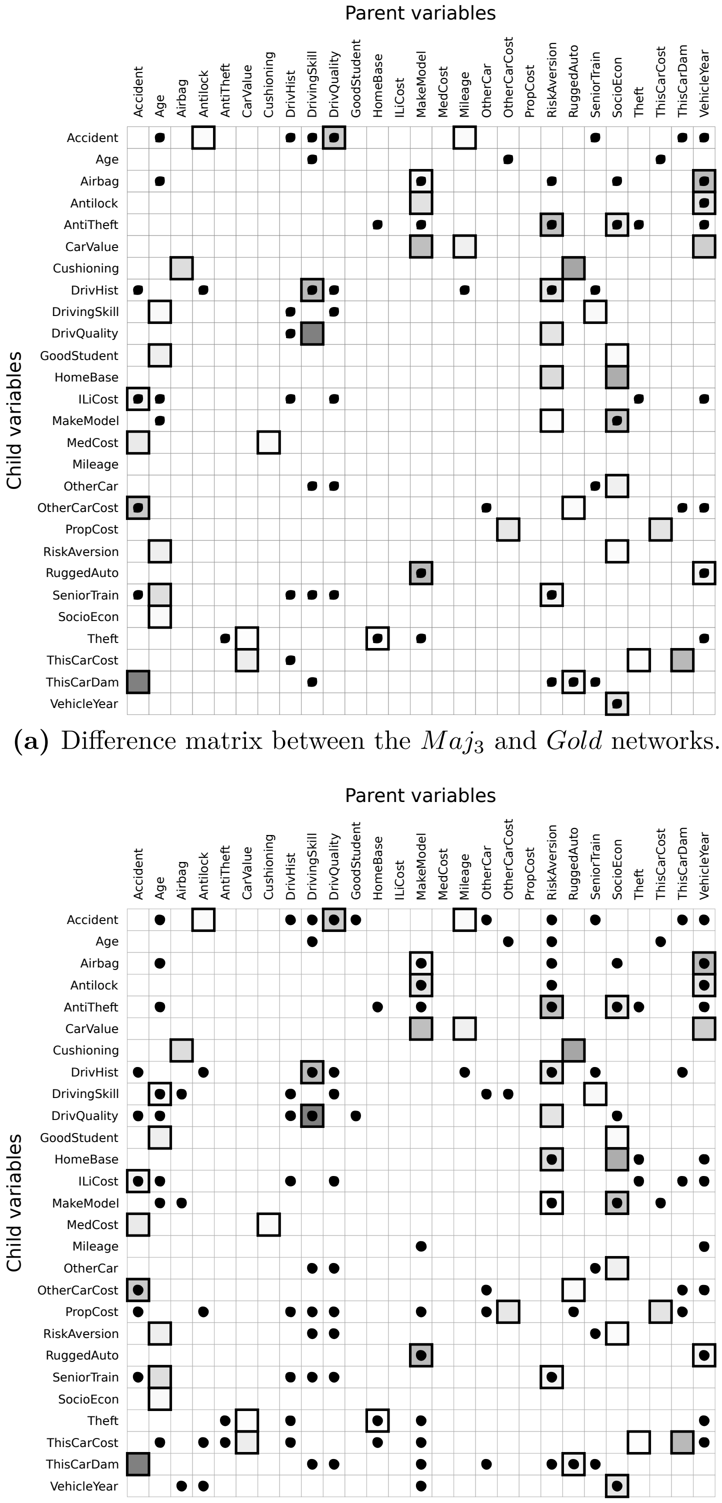
The two difference networks shown are the two of the better Maj and DS network respectively, as measured in Section 5.3. The $DS_{0.15}$ difference matrix in Figure 5.8b includes a slightly higher proportion of strong arcs from the *Gold* network which were not present in the Maj_3 network. Most notable is the *DrivingSkill* → *DrivQuality* arc. However, this comes at the expense of many more false positives in the $DS_{0.15}$ network structure which are not in the *Gold* network.



(a) Summary matrix for the Maj_3 network. The lightest cells were included by three of the six allocated participants, and the darkest were included by all six.

(b) This summary matrix shows the strength of each arc as defined by the EM algorithm for the $DS_{0.15}$ network structure. Arcs which are included despite a lower prior probability of including arcs are darker. Those not included until the prior probability for any given arc is high are lighter.

Figure 5.7.: Summary of Maj_3 and $DS_{0.15}$ network structures.



(a) Difference matrix between the Maj_3 and $Gold$ networks.

(b) Difference matrix between the $DS_{0.15}$ and $Gold$ networks.

Figure 5.8.: These difference matrices show the Maj_3 and $DS_{0.15}$ arcs in black circles and the arcs from the $Gold$ network in squares.

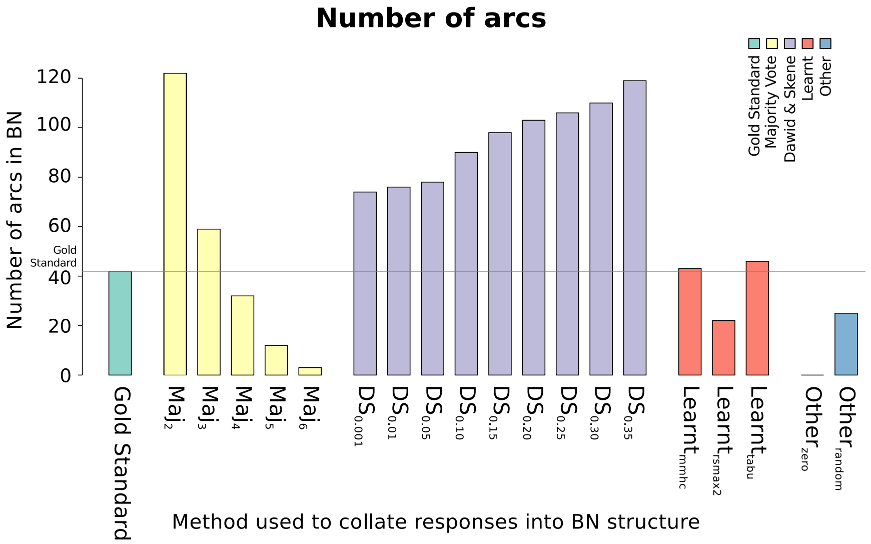


Figure 5.9.: Number of arcs in the *Gold* and *Eval* networks (see Appendix C, p254, for full data).

5.2.4. Number of Arcs in Network Structures

Figure 5.9 shows the number of arcs in the *Gold* and each of the *Eval* network structures. It is desirable for networks to have a similar number of arcs as the *Gold* network, to give it the best shot of encoding the same causal relationships as that network.

From Figure 5.9, there is a clear progression in the *Maj* and *DS* structures where the arcs decline with a higher threshold for what defines a majority, or a lower prior for whether an arc exists or not in the EM algorithm.

5.3. Scoring the Evaluation Networks Against the Gold Standard

The previous section provided an overview of the *Eval* networks and how they were constructed. This section will take the *Survey* networks and compare them to the *Gold* network using various metrics to evaluate how “good” they are. There are both qualitative and quantitative ways to evaluate a BN, with this research performing primarily a quantitative evaluation.

With respect to quantitative evaluation, there is one overriding goal: Does the network represent the same probability distribution as the original, and if not, how different is it? This is important because one of the primary reasons that a BN is used in problem solving is to factorise a complex probability distribution into one which is more computationally tractable and understandable.

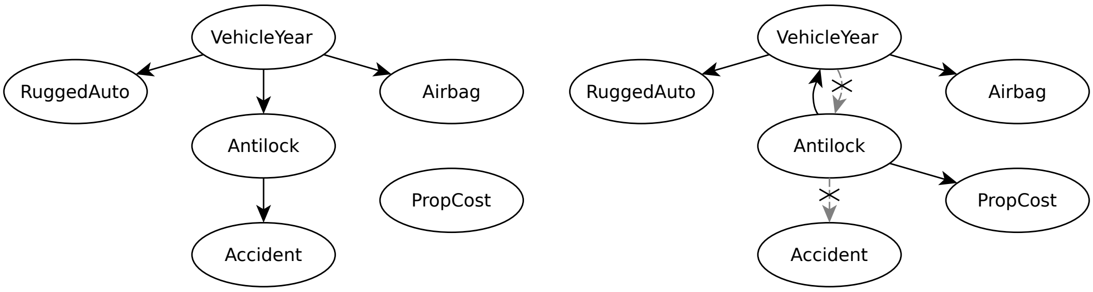
A somewhat more qualitative goal is to measure if the BN has the same structure as the gold standard. Although vastly different structures can depict similar probability distributions it is still a useful metric of “goodness”. For example, if the structure of the original network was learned from data, then it is unlikely that the arcs represent causal relationships. However, when using traditional KEBN or SEBN then it is more likely that the edges represent causality (Korb and Nicholson, 2011, p311). It turns out that the insurance BN used for evaluating this research is mostly causal. Thus, it does make sense to compare the structures of the *Survey* network structures with the original, to see how close they are.

The first metric used is the Structural Hamming Distance, which is a measure of the qualitative similarity between the network structures (Section 5.3.1). Following this is an investigation into the Receiver Operating Characteristic and F1 score (Section 5.3.2) which are elaborations of the SHD that take into account whether incorrect arcs are false positives or false negatives. Section 5.3.3 then moves onto a more quantitative analysis of the networks, by parameterising them and then seeing how close the probability distribution encoded by each evaluation network compares to the *Gold* network. Given the poor results obtained by each of the criteria described above and discussed below, Section 5.3.4 investigates further to see what would happen to the results if less participants were participating, or if they provided noisier responses. The goal is to get some insight into whether the situation would reasonably be expected to improve if more participants were added, or if participants were asked to answer more questions resulting in more data.

5.3.1. Scoring Networks Using the SHD Metric

The Structural Hamming Distance (SHD, Tsamardinos et al. (2006), Figure 5.10), also referred to as the edit distance (Flores et al., 2011), can be described as the number of operations that need to be performed to make two DAGs match (Tsamardinos et al., 2006). As its name suggests, it is derived from the term "hamming distance" which describes the number of characters which need to be changed to go from one string to another (Hamming, 1950). When calculating the SHD of two BNs, the operations which can be performed to transform one network structure into the other (shown in Figure 5.10) are:

- Remove edge
- Add edge
- Reverse edge



(a) Snippet of the *Original* BN with six nodes and four arcs.

(b) Similar snippet of the $DS_{0.10}$ BN.

Figure 5.10.: Example of two network fragments with a SHD of three. Notice the addition of *Antilock* → *PropCost*, the removal of *Antilock* → *Accident*, and the reversal of *VehicleYear* → *Antilock*.

The SHD is a useful, although not definitive metric for comparing the structure of two networks. It is not definitive, because it is still possible to have a BN model accurately represent the same probability distribution, even though it has a different structure. However, it is useful, because if both networks are elicited from experts and the arcs are meant to represent causal relationships, then the omission or addition of a causal relationship in one model does provide feedback about how similar the models are. Given that the gold standard network had its structure elicited by experts (Section 5.1.3, p105), it would be ideal if the structure elicited from this evaluation was similar. It would go to show that when using SEBN, even though participants are only asked questions regarding a small subset of possible relationships in the network, the overall structure after collating results is similar. Alternatively, if the structure was completely different and no arcs were the same, then that would provide evidence that SEBN tends to force people to think in different, perhaps unintuitive, ways (Proposition 8, p66). The difference matrices in Section 5.2.3 visualised this metric, but this section will quantify and investigate it in further detail.

How Does SHD Impact on Other Scoring Metrics?

In order to check the impact of the SHD on the overall "quality" of the network, or perhaps better termed, the ability of the network to represent the same distribution that a data set was drawn from, additional analysis will be performed. This analysis will not relate to the *Survey* networks specifically. Rather, it will take the *Gold* network structure and see how it responds to random permutations which increase the SHD from its unmodified state.

This analysis was conducted by iteratively performing a random operation on the network structure (add/remove/reverse arc) that doesn't introduce a cycle. This augmented network was parameterising using 500 data points sampled from the original network using the `rbn` function from the *bnlearn* R package (Scutari, 2010), then scoring against the original using the BIC criterion.

Figure 5.11 shows that as the number of random perturbations (and hence the SHD) *increases* the network quality *decreases*. This shows that a higher SHD tends to have a detrimental impact on the ability to represent a similar probability distribution as the *Gold* network.

SHD of Networks Elicited During Evaluation

The SHD of all *Eval* networks is shown in Figure 5.12. It is apparent that the distance from the *Maj* and *DS* networks to the original is quite far. It also

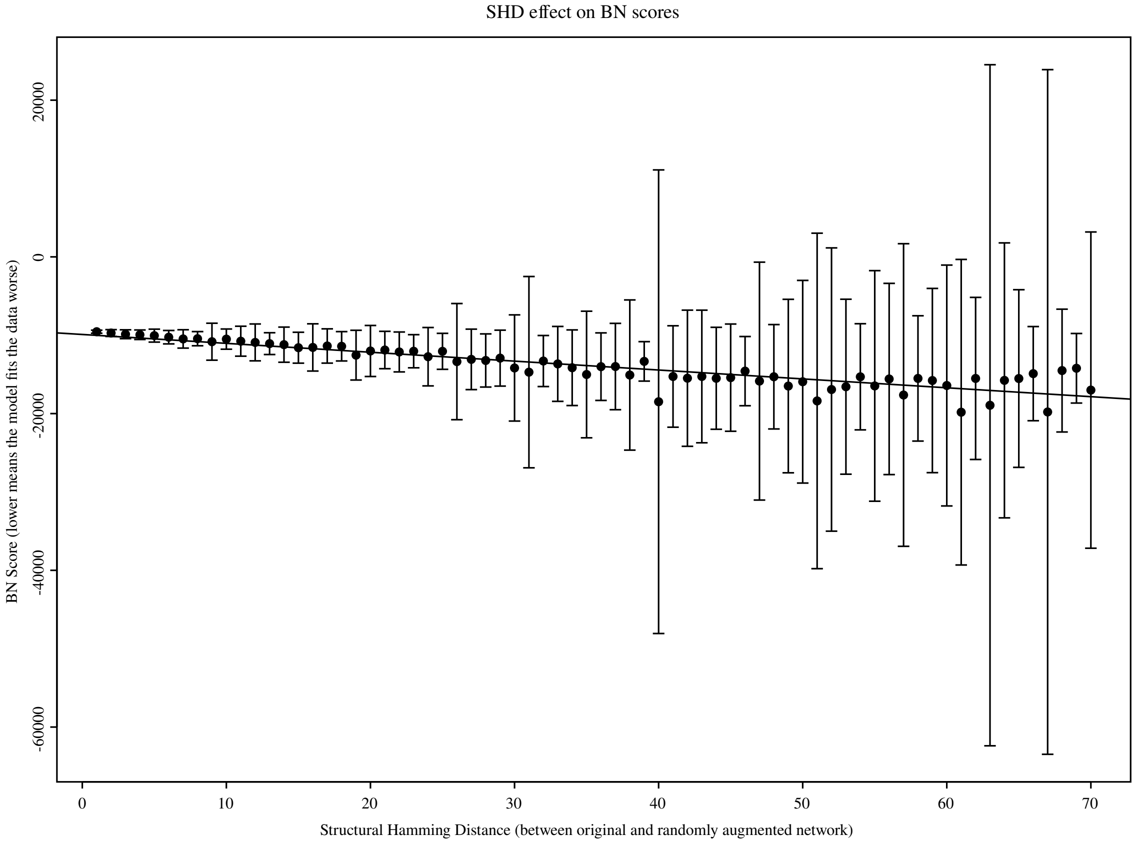


Figure 5.11.: As a BN structure diverges more, the ability of it to fit a particular distribution degrades. The error bars show the minimum and maximum scores achieved over 100 randomly augmented networks with the same SHD value.

degrades as less people are required to agree in order to form a majority vote or as the Dawid & Skene prior is increased.

The *Learnt* networks all exhibit a comparatively low SHD. This is surprising, given they are not interested in causal relationships, and so have different heuristics to guide whether an arc should exist or not. Given this fact, it would be expected that the SHD of the *Survey* networks would be lower (closer in structure to the *Gold* network).

The SHDs of the *Other* networks are not very interesting. $Other_{zero}$ will always have an SHD equal to the number of arcs in the *Gold* network. The SHD of the $Other_{rand}$ arc depends on the algorithm being used to create the random structure.

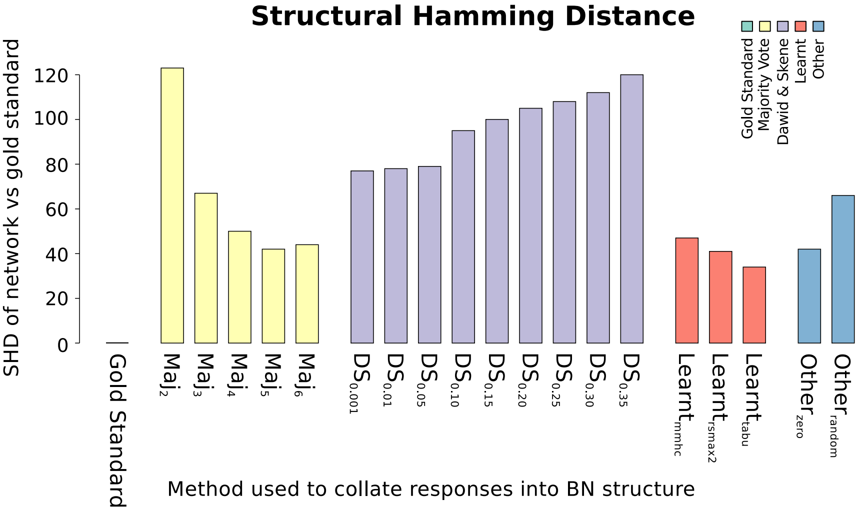


Figure 5.12.: SHD of the *Eval* networks compared to the *Gold* network (see Appendix C, p255 for full data).

The fact that *Maj* networks with low thresholds, or *DS* networks with high priors have such large SHD scores is because they tend to include so many arcs. Although the SHD score doesn't distinguish between false positives or false negatives, the high SHDs tend to be due to false positives rather than false negatives. This is investigated further in Section 5.3.2.

5.3.2. Scoring Networks Using the ROC and F1 Metric

The Receiver Operating Characteristic (ROC) is a tool with which to compare various algorithms, and decide which is best. It makes use of similar metrics to the SHD. The metrics it uses in order to compare algorithms is the True Positive Rate (TPR, Eq. 5.1) and the False Positive Rate (FPR, Eq. 5.2) framed in terms of the number of True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN):

$$
\begin{aligned}
TPR &= \frac{TP}{TP + FN} && (5.1)\\
FPR &= \frac{FP}{FP + TN} && (5.2)\\
F_1 &= 2 \times \frac{Precision \times Recall}{Precision + Recall} = \frac{2 \times TP}{2 \times TP + FN + FP} && (5.3)
\end{aligned}
$$

A perfect algorithm would have a TPR of 1 (all arcs in the *Gold* network were included in the evaluation network) and a FPR of 0 (there are not any arcs in the evaluation network which are not found in the *Gold* network)[8]. In most cases, it becomes a trade-off between high TPR and low FPR as to which algorithm is better. In some situations, a higher TPR may be desirable (e.g. in disease diagnosis - where missing a positive result can cause serious problems). Other times, it might be preferable to have a lower FPR (if the decisions being taken in response to the algorithm cost significant money, one may not wish to spend that money on something which turns out to be a false positive). Often, as an algorithm approaches a higher TPR, the number of false positives increases asymptotically.

Given that the number of CPT values required for a variable increases exponentially with the number of parents it has in a BN, many algorithms for learning BN structures optimise for a lower number of parents (e.g. Wallace and Korb (1999) explicitly favours simpler models). Thus, it makes sense that the ROC metric for the evaluation network structure should prioritise a lower FPR. Having said this, it should not be completely at the expense of true positives. In the extreme case, if there were no true positives, then the network structures would have none of the same arcs at all.

Figure 5.13 shows the ROC “space” for the *Eval* and *Gold* networks. The *Gold* network is, as expected, the best, given that by definition it has a TPR of 1 and a FPR of 0. After this, the $Learnt_{tabu}$ network stands out as much closer to the *Gold* network. This is primarily because the TRP is better than all other *Eval*

[8] This concept of a “perfect” algorithm is not quite right. False positive of 0 and false negative of 0 is probably indicative of heavily over-fitting the problem, but it is a good thing to strive for in general.

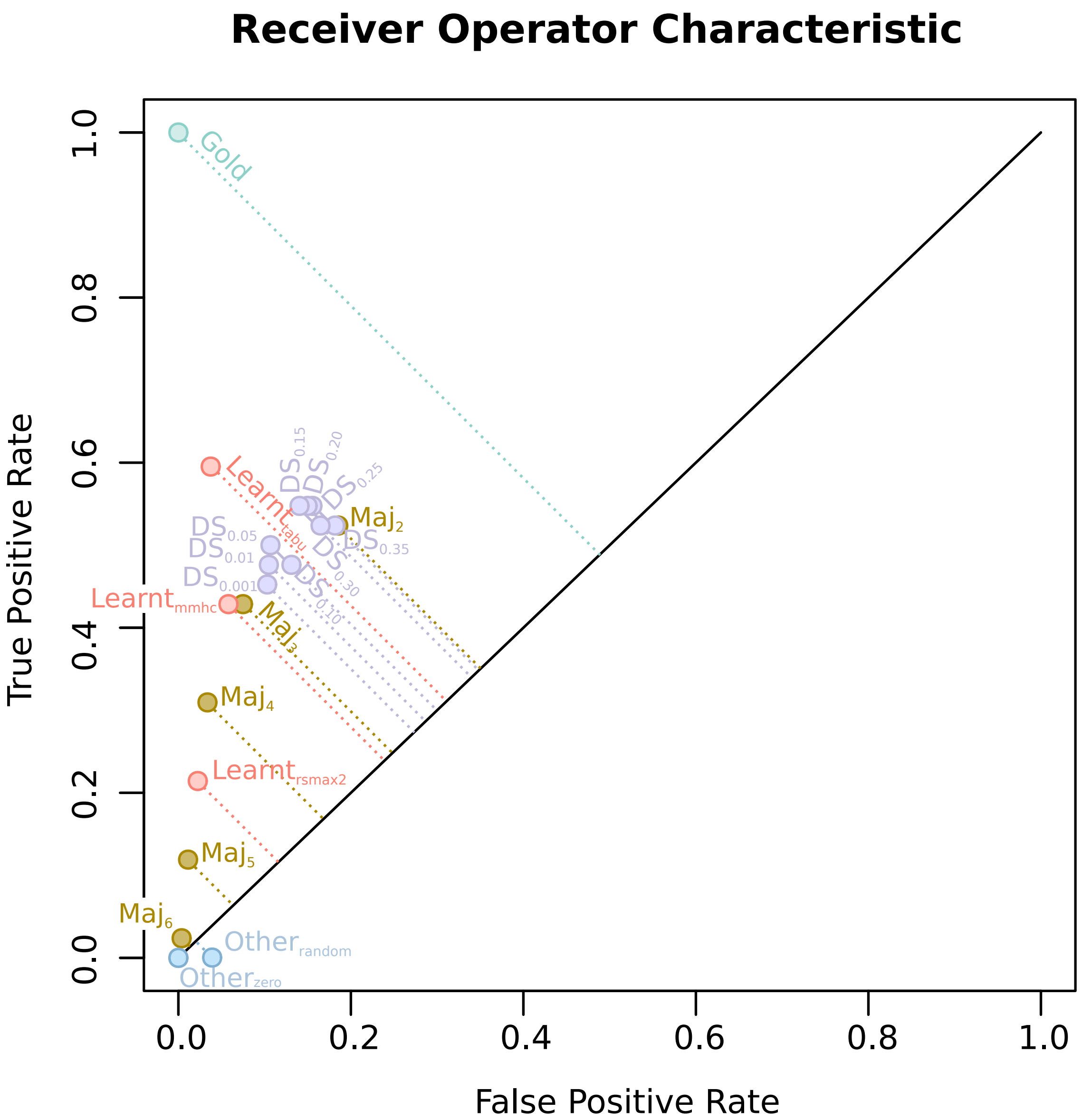


Figure 5.13.: The ROC space of arcs. Those closer to the top left are better, because they are "more correct" (higher TPR) and "less incorrect" (lower FPR).

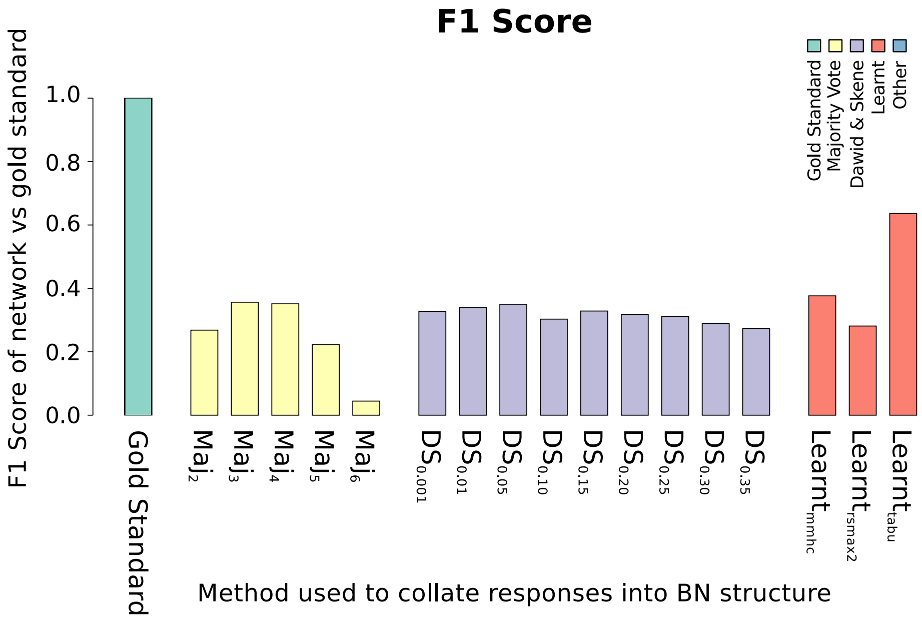


Figure 5.14.: F1 score for the *Eval* networks when compared to the *Gold* network.

networks. Even though there are some networks with a very low FPR, this is due to the fact that they have so few arcs compared to the *Gold* network.

In terms of the *Survey* networks, the *DS* networks tend to outperform the *Maj* networks. Although $Maj_{2,3}$ are both close to the *DS* networks, they are outperformed by nearly all of them. The remainder are at least as good if not better, while $Maj_{4,5,6}$ have too few arcs to be considered close to *Gold* by the ROC metric.

Another metric related to the sensitivity and specificity measured by the SHD is the F1 score (Eq. 5.3). This is a single metric which takes into account both precision and recall providing a single number which is the harmonic mean between the two. This is ideal because to get a high F1 score (i.e. near 1) both the precision and the recall need to be good. As can be seen in Figure 5.14, many of the *Survey* BN structures outperform $Learnt_{rsmax2}$, however $Learnt_{mmhc}$ and particularly $Learnt_{tabu}$ greatly outperform each *Survey* structure. It can also be seen that the best *Maj* network structure is similar to the best *DS* structure, although the *Maj* scores degrade quickly with higher thresholds.

Post-hoc Analysis of SHD results Another clear benefit of DS over Maj which is apparent in Figure 5.13 and Figure 5.14 is that whereas Maj networks reduce their TPR and $F1$ scores as the threshold increases, DS networks scale much less drastically. What this means in practice is that when eliciting BNs in the absence of a gold standard, a knowledge engineer needs to make a judgment call about the right threshold for majority vote. If the judgment call is a little bit off (e.g. 5 instead of 4 in this evaluation) then the quality of the BN could degrade significantly. However with the DS networks, the penalty for choosing the incorrect threshold is less severe. This should be investigated in future studies to see if it is the case with other BNs.

5.3.3. Scoring Networks by Parameterising Then Using the BIC Metric

Many techniques exist to measure the accuracy with which a given BN models a particular data set. The log likelihood (LL) metric is used to measure the probability that a particular data set would be observed, if the given BN did indeed represent the true probability distribution. The observed data set may have been collected from experiments, sampled from an already existing BN, or obtained in some other manner. The main problem with only using the LL is that models which over-fit the solution are ranked highly. To address this issue, algorithms such as the Bayesian Information Criterion (BIC), Akaike's Information Criterion (AIC, Akaike, 1998), and the Minimum Description Length (MDL, Rissanen, 1978) have been proposed to balance the desire for accurate measurements with the desire to have simple models that don't over-fit the problem (Cruz-Ramírez et al., 2006). These algorithms penalise models which have larger number of parameters. The BIC score is used to score the parameterised networks in this section.

The collated network structures from Section 5.2 need to be parameterised before comparing to the *Gold* network. In order to do this a data set to use for parameterising the network is required and another (ideally different) data set for scoring the network. Both of these can be sampled from the *Gold* BN which formed the basis for the new network being elicited (Figure 5.15).

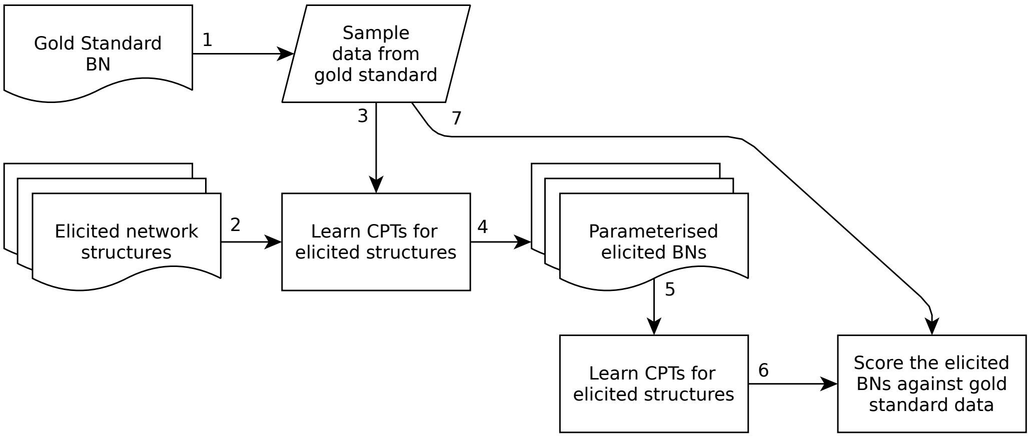

Gold Standard BN
1
Sample data from gold standard
3
7
Elicited network structures
2
Learn CPTs for elicited structures
4
Parameterised elicited BNs
5
Learn CPTs for elicited structures
6
Score the elicited BNs against gold standard data

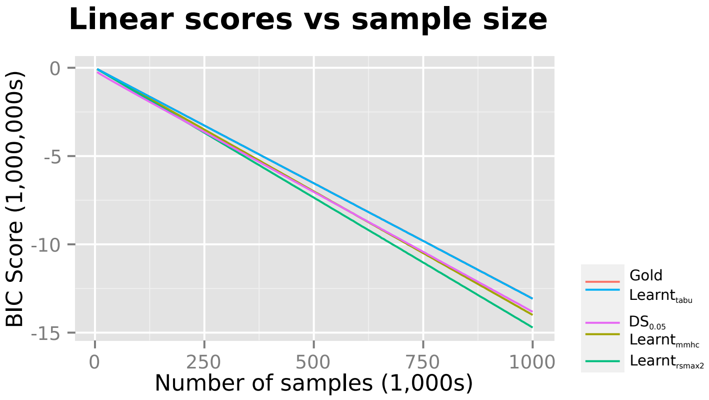
Figure 5.15.: Process for scoring elicited network structures when they do not yet have CPTs. Note that the data used in step 3 for learning CPTs, and that used for step 7 to score the evaluation network are both different data sets, randomly sampled from the same gold standard BN.

In order to score the *Survey* network structures, the *bnlearn* (Scutari, 2010) package for the R statistical software *(R Core Team, 2014)* was used. A synthetic data set of 1,000,000 items was generated using the `rbn` function, which samples from the distribution represented by the *Gold* network.

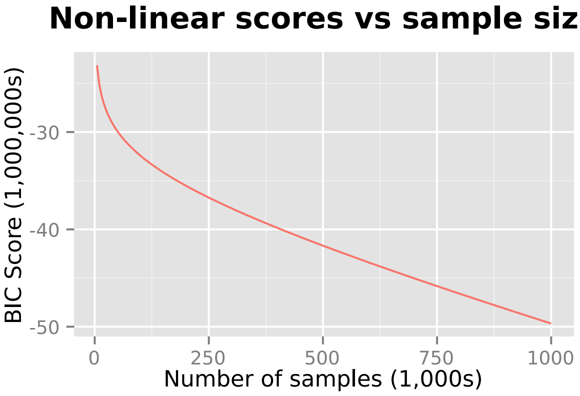
The *Survey* network structures were then input into the package. Using these network structures and also the synthetic data set from the previous step, the CPTs were learnt using the `bn.fit` function of *bnlearn*. To compare these to the *Gold* BN, the `bic` scoring function from the *bnlearn* package was used. The *Learnt* and *Other* networks were also scored to put the *Survey* scores into perspective.

BN Score vs Size of Data Set Used to Calculate Probabilities Before proceeding to score the *Eval* networks against the *Gold* network it is important to know how many samples should be used to parameterise the networks. Figure 5.16 shows the score that is given to five different car insurance BNs, as they are parameterised with an increasingly large number of observations. This analysis was done by sampling increasingly large data sets from the insurance network, using the `rbn` function from *bnlearn*.

As the number of samples grows larger, it becomes less likely that the specific data



(a) BIC scores for *Gold*, *Learnt*, and $DS_{0.05}$ networks, after being parameterised with progressively larger sample sizes drawn from the *Gold* network itself.



(b) BIC scores for the $DS_{0.35}$ network, showing that networks with a larger number of arcs tend to follow a logarithmic rather than linear change in the BIC score, until there is enough data to model all of the relationships in the network.

Figure 5.16.: BIC score of various BNs, after being parameterised with progressively larger sample sizes.

set came from a particular model (the BIC falls). This is due to an increasing amount of information in the data set, but a uniform number of variables and states in the BN able to encode that extra information. As a result, it is harder for a BN to model the nuances of 500,000 data points than it is to model 5,000.

It can also be seen that although most networks BIC scores fall as good as linearly (Figure 5.16a), the $DS_{0.35}$ score initially falls logarithmically, until there is enough data for which to parameterise all of the extra additional arcs (Figure 5.16b). After that point, it too seems to fall linearly[9]. When conducting analysis of BN structures, it is therefore important to ensure that the amount of data being used to parameterise the structure is great enough to ensure it is in the linear portion of this graph. Otherwise, the amount of data used will greatly impact the relative score of a particular structure. For example, when comparing BIC scores for the *Gold* , $Learnt_{mmhc}$, and $DS_{0.35}$ networks, then the relative difference between 25k to 50k data points will be far greater than 525k to 550k.

Comparing DS, Maj, *Learnt*, and *Other* scores Figure 5.17 shows the scores assigned to the DS and Maj network structures when parameterised with 1,000,000 data points sampled from the *Gold* BN. For comparison, it also shows the *Gold*, *Learnt* and *Other* network scores.

It can be seen that the Maj networks exhibit a sudden decline in quality as the threshold for agreement drops from 3 to 2. If the threshold is reduced, then more arcs will be included, and it seems that a majority value of 2 was enough to produce a poor network. However in the other direction, it only slowly degrades as the threshold is increased from 3 to 6. This degrade in quality will be due to the fact that there are less arcs, resulting in an inability for the BN to be able to encode the relevant information.

For the DS networks, as the prior probability of an arc increases (increasing the chance of any single arc being included in the final structure) then the score worsens. As the prior approaches zero, then the probability of any given arc being

[9] The $DS_{0.35}$ network was chosen to highlight this non-linear relationship because it was the *Eval* network which showed this most clearly. $DS_{0.30}$ and lower showed the same pattern, but less so, to the point where $DS_{0.20}$ was close to linear.

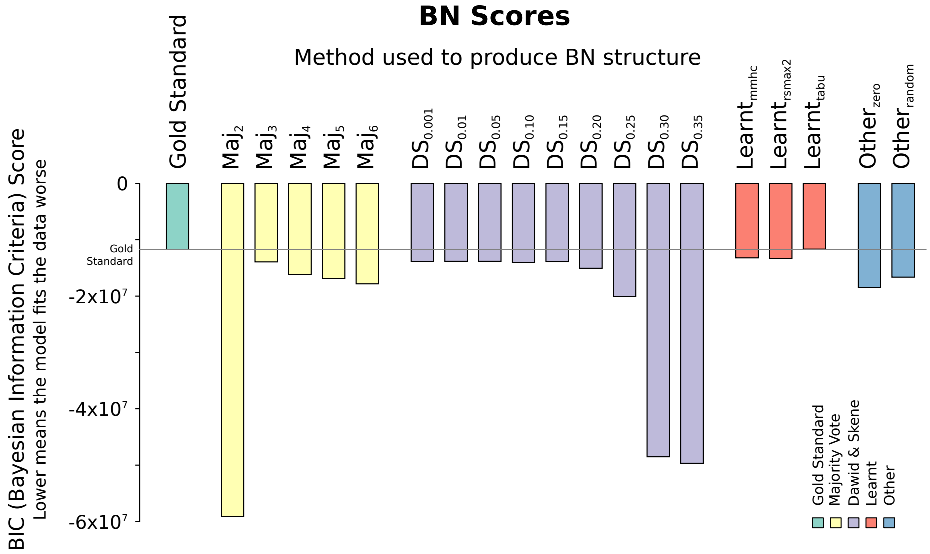


Figure 5.17.: Scores calculated for the *Gold* and *Eval* networks (see Appendix C, p253 for full data).

included decreases and the total number of arcs is reduced. As the prior probability of any given arc being included approaches 1.0, then almost all arcs from Maj_1 will get included, regardless of how many people rejected it.

Conclusion of BIC score analysis The BIC score of all *Eval* networks was worse than both $Learnt_{mmhc}$ and $Learnt_{tabu}$. The best Maj network was Maj_3, and the best DS network was $DS_{0.01}$ which each had the same score, closely followed by $DS_{0.001}$ and $DS_{0.05,0.10,0.15}$ that had similar scores.

Note that Maj_2, and $DS_{0.30,0.35}$ are so far worse than any other structure, including *Zero* and *Random* that they are not even worth considering as potential structures. Given that $DS_{0.25}$ and $Maj_{4,5,6}$ are near enough to the *Other* networks that they are not useful. They are unable to provide a better model of the car insurance domain than either random arcs between variables or zero relationships at all.

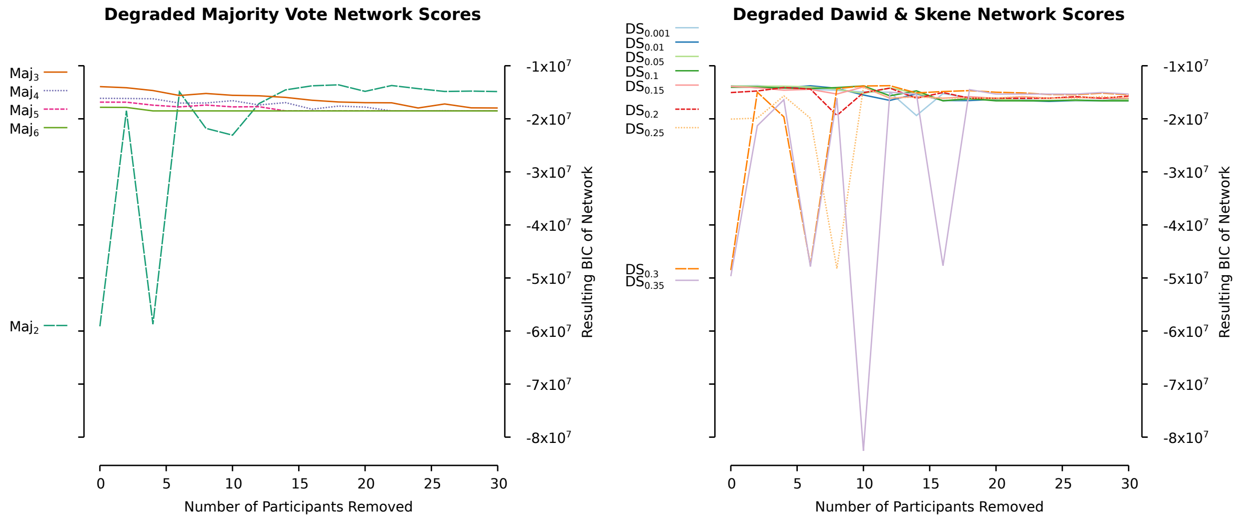
5.3.4. Intentionally Degrading Survey Response Data (Post-Hoc)

The analysis to this point has shown poor results in that the *Survey* networks are not ideal and the *DS* networks are not substantially better than the *Maj* networks. One limitation of this evaluation is that there was only one survey conducted due to time constraints. This post-hoc analysis section will investigate what happens when responses belonging to an increasingly large proportion of participants are removed from the survey responses before analysis. This will simulate running the survey again with less participants. As less participants are included, trends can be established as to how EM and Majority Vote are able to cope with less data.

Past research in the field of crowd sourcing has made use of different numbers of responses. Whitehill et al. (2009) acquired three labels for each data point, whereas Sheng et al. (2008) explicitly investigated the effects of more participants. Given the results of Sheng et al. (2008) which showed that the improvement in results diminishes with an increased number of participants, it may be that in this evaluation study, allocating each question to 6 participants reached a saturation point. As a result, the EM algorithm might not have had enough of a chance to differentiate itself from the Majority Vote. Randomly removing participants from the response pool may provide an insight into what to expect if less experts are available.

Figure 5.18 and Figure 5.19 show the result of this analysis. The BIC score of all *Eval* networks degrade as more participants responses are excluded from the collation stage. However, the *DS* structures are affected less and converge on BIC scores between -1.5×10^7 and -1.7×10^7 (Figure 5.19b). Comparatively, the *Maj* networks converge on worse scores between -1.8×10^7 and -1.9×10^7(Figure 5.19a). Note that the score which $Maj_{4,5,6}$ converge on is that of $Other_{zero}$ as they very quickly become networks with zero arcs.

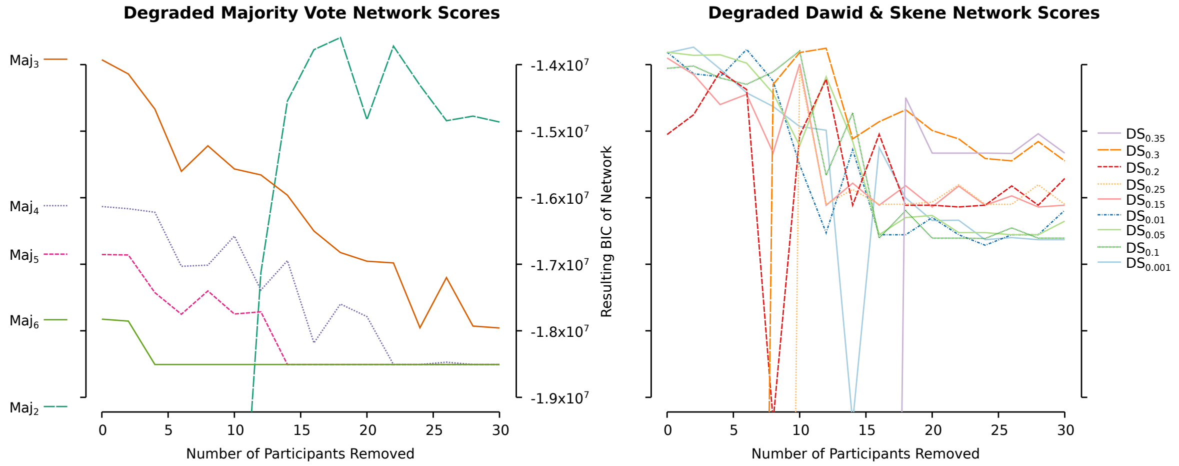
5.3.5. Summary of "Good" Networks



(a) *Maj* network scores after removing random participants.

(b) *DS* network scores after removing random participants.

Figure 5.18.: *Eval* network scores after removing between 0-30 random participants before collating structures.



(a) *Maj* network scores after removing random participants.

(b) *Maj* network scores after removing random participants.

Figure 5.19.: Cropped view to better show *Eval* network scores after removing between 0-30 random participants before collating structures.

Network	SHD	F1	BIC	Degraded BIC
Maj_2	14th	12th	14th	1st
Maj_3	4th	1st	5th	11th
Maj_4	3rd	2nd	8th	12th
Maj_5	1st	13th	9th	13th
Maj_6	2nd	14th	10th	14th
$DS_{0.001}$	5th	6th	3rd	10th
$DS_{0.01}$	6th	4th	1st	7th
$DS_{0.05}$	7th	3rd	2nd	8th
$DS_{0.10}$	8th	9th	6th	9th
$DS_{0.15}$	9th	5th	4th	6th
$DS_{0.20}$	10th	7th	7th	4th
$DS_{0.25}$	11th	8th	11th	5th
$DS_{0.30}$	12th	10th	12th	3rd
$DS_{0.35}$	13th	11th	13th	2nd

Table 5.3.: Summary of the evaluation metrics in Section 5.3, comparing *Eval* networks to the *Gold* network. Each *Eval* network is ranked from 1st to 14th. Metrics which were considered unacceptable relative to the other networks are greyed out.

A summary of Section 5.3 is shown in Table 5.3, depicting the various metrics which were used to score the *Eval* BN structures. Section 5.3.1 concluded that all *Eval* networks had a poor SHD score, however the Maj networks performed better than DS due to the excessive amount of false positives in DS network structures. The F1 score of $Maj_{3,4}$ were better than each of the DS networks, although the remaining $Maj_{2,5,6}$ had particularly poor F1 scores. $DS_{0.01,0.05,0.001,0.15}$ produced the best BIC scores followed by Maj_3. All DS networks were better able to handle less experts than each of the Maj networks except Maj_2.

5.4. Time Spent by Participants Doing Survey

The BNE software measured the duration spent by each participant conducting the online survey. This was done by analysing logs which indicate when specific actions were performed such as signing in or answering a question. This is similar to the approach taken by most web analytics companies in practice. Despite its

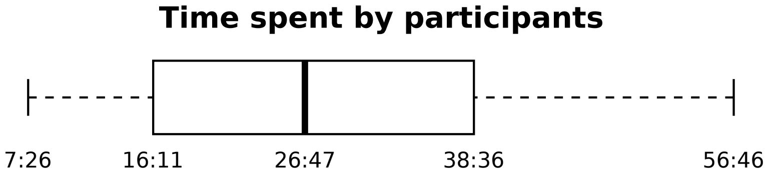


Figure 5.20.: Analysis of time spent by participants using the survey software, who completed all questions allocated to them in the first round, shown as quartiles.

popular usage, measuring time spent on websites using logs is problematic. At the time of writing, Google Analytics arbitrarily defines a "session" as a time frame where a sequence of actions were taken with no more than 30 minute breaks between each action. Thus if somebody was to perform an action then leave for 31 minutes before returning, it would count as two different sessions. However if that same user only left for 29 minutes, it would count as one continuous session[10]. In the future, it may be preferable to modify BNE to use JavaScript to monitor mouse movements, touches, and key presses on each web page, although this would not work appropriately for mobile browsers that use touch screens.

Figure 5.20 provide an insight into the approximate amount of time spent by participants in order to elicit the structure of a BN. The minimum amount of time spent was less than 10 minutes, the median was under 30 minutes, and the most amount of time spent was just under 1 hour.

Milton (2008, p50) explain that traditional KE projects typically last between 4 to 24 weeks, can require "a few hours per week ... from each expert" (p25), and could include "zero, one, two, or many" experts (p50). At the lower end of this scale, this would involve approximately 8 hours from a single expert over a 4 week project. In a larger project, it may be closer to 100 hours for a 24 week project with two experts.

Milton (2008) describe an entire process from conceiving the idea to build a model, through to completion and usage of the build model. It must be noted that this evaluation study which output a BN structure was at most $\frac{1}{3}$ of an entire project, given most BN projects will need to elicit variables and CPTs in addition to the

[10] `https://support.google.com/analytics/answer/2731565?hl=en`

structure. However, it still required substantially less time from each participant who took part compared to usual KA projects.

5.5. Comparing Estimated Quality Measures to Accuracy

Once the survey was completed and the evaluation BNs constructed, the expertise of each participant was estimated using a technique described by Ipeirotis et al. (2010). The real accuracy of the participants was also garnered by comparing their survey responses to the gold standard network ($\frac{TP+TN}{Total\,questions\,asked}$). Comparing the estimated and real accuracy verified whether or not SEBN was able to correctly estimate expertise of different participants (Figure 5.21).

The Dawid & Skene algorithm was used to decide which arcs to include in the network based on participant responses. Ipeirotis et al. (2010) show how the EM algorithm can also be used to compute the "quality" of each participant. The quality refers to how consistent their responses are with the information in the *Gold* network. The BNE software calculated the estimated quality using the software from Ipeirotis et al. (2010). Figure 5.22 shows each participant's accuracy compared to the estimated quality for the $DS_{0.05}$ network structure. It can be seen that for this evaluation, the estimated quality is *not* a good indicator of a

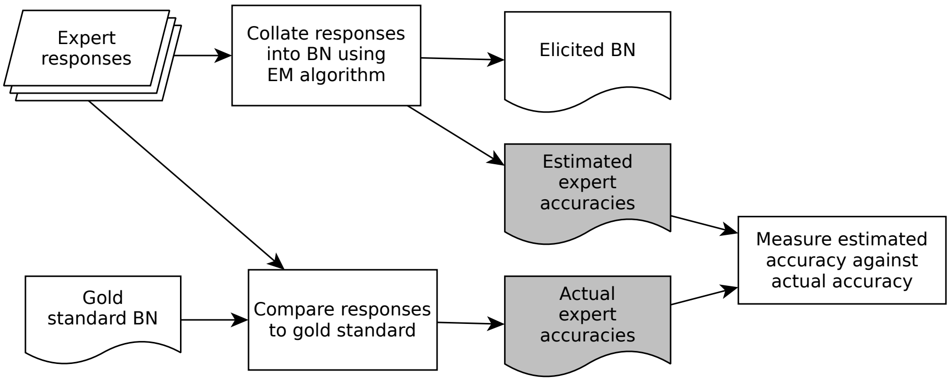

Expert responses
Collate responses into BN using EM algorithm
Elicited BN
Estimated expert accuracies
Measure estimated accuracy against actual accuracy
Gold standard BN
Compare responses to gold standard
Actual expert accuracies

Figure 5.21.: Process for evaluating whether SEBN is able to meaningfully discern the expertise of each participant or not.

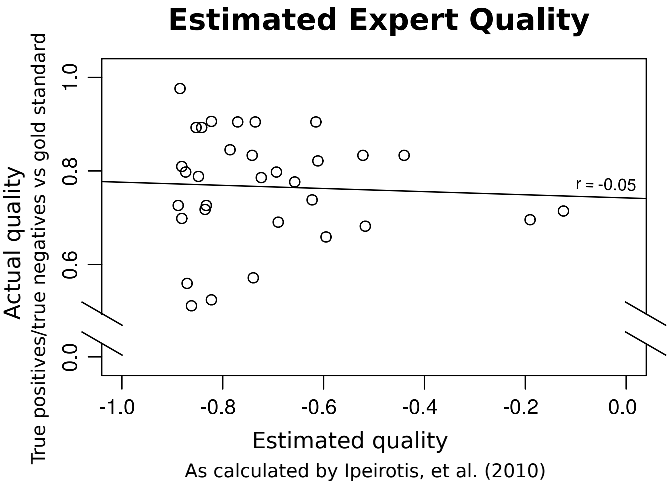


Figure 5.22.: The quality of participants responses as per the Ipeirotis et al. (2010) calculation compared to the actual accuracy measured from *Gold* vs $DS_{0.05}$ network.

participant's accuracy. The future work in Section 8.7.6 (p217) discusses different algorithms that could be used to estimate the quality of participants in future studies.

5.6. Optimising Network Structure for Future CPT Elicitation

Optimisations which can be performed once the network structure is collated were discussed in Section 4.6.3 (p97). These optimisations are not related to making the process of *structure* elicitation more efficient or effective, but rather to help with subsequent *CPT* elicitation.

Although the evaluation survey didn't result in any optimisation opportunities, the time required of participants to ask about optimisations is negligible. If there are no potential optimisations as is the case with the *Survey*/*Gold* network, then there is no additional time required. Therefore, this section will investigate some hypothetical scenarios from some of the *Eval* networks by making changes to the network structure and then analysing how applying specific optimisations would have reduced the required time of experts and knowledge engineers alike.

5.6.1. Changes to $Eval$ Networks to Facilitate Evaluation of Optimisations

The evaluations discussed were the NoisyOR and NoisyMAX. NoisyMAX is potentially suitable for ordinal nodes with multiple ordinal parents and NoisyOR for boolean nodes which have multiple boolean parents.

Modifying Maj_3 for NoisyMAX Optimisations

The nodes which are chosen for modification and subsequent optimisation from the Maj_3 network are those which are ordinal, have a large number of parents, and most of the parents are ordinal. Nodes with more parents will benefit more from optimisations, due to the way in which the number of CPT parameters required increases exponentially with the number of parents.

Maj_3 contains four ordinal nodes which have ≥ 5 parents, for which only one is not ordinal (Figure 5.23). The non-ordinal parents (greyed out in Figure 5.23) were removed for the purpose of this analysis.

Modifying $DS_{0.10}$ for NoisyOR optimisations

As with the previous section, there are none which fit into the criteria of being boolean and having only boolean parents. The boolean node closest to having all boolean parents is *Theft* (Figure 5.24). This variable has five parents in $DS_{0.10}$ of which *VehicleYear* and *AntiTheft* are boolean. In addition, *DrivHist* will be considered boolean by changing the states in allows from *{ Zero*, *One*, *Many }* to *{ Zero*, *Many* }. The resulting relationship after removing *MakeModel* and *HomeBase* is *{ DrivHist, VehicleYear, AntiTheft }* → *Theft*.

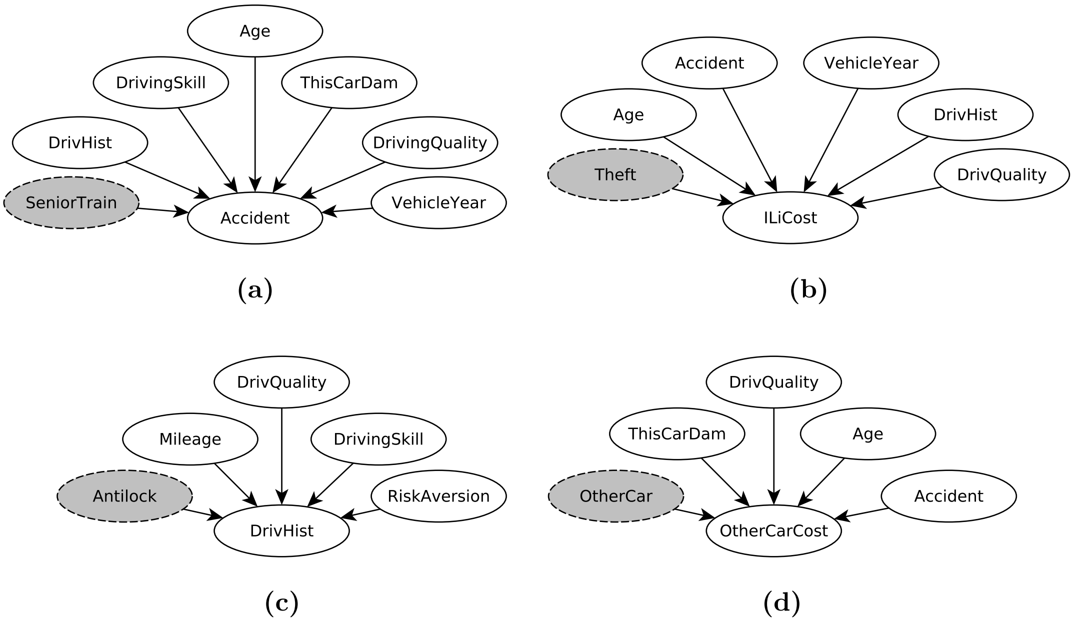


Figure 5.23.: Four ordinal variables from Maj_3 with ≥ 5 parents of which only one is not ordinal. The non ordinal parent is shown with a shaded background.

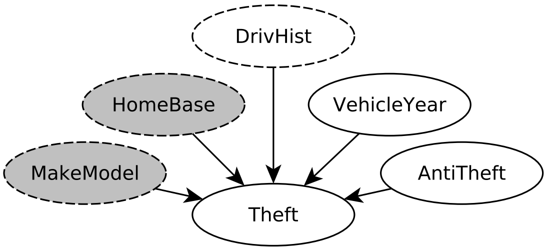


Figure 5.24.: Theft variable from $DS_{0.10}$ with five parents of which three will be considered boolean for this analysis. The non boolean parents are shown with a shaded background.

Child Node	Number of parameters required for CPT		
	Regular CPT	NoisyMAX	Parameter Reduction (% of original parameters)
OtherCarCost	576	18	3.16%
Accident	$2,592$	22	0.84%
ILiCost	864	19	2.20%
DrivHist	432	17	3.94%

Table 5.4.: Reduction in parameters required to parameterise random variables by using the NoisyMAX distribution.

5.6.2. Parameter Reduction When Applying NoisyMAX Optimisation

Table 5.4 shows the reduction in CPT parameters requiring eliciting for nodes that the NoisyMAX optimisation can be applied to. All of the four variables reduce by at least one order of magnitude, and in the case of *Accident*, by two orders of magnitude. It is not feasible to ask experts to elicit over 2,500 probabilities to parameterise a single node, when there are 10s of nodes requiring parametrisation. However, eliciting 22 probabilities is much more reasonable.

5.6.3. Parameter Reduction When Applying NoisyOR Optimisation

When applying the NoisyOR optimisation to the *Theft* node reduces the required number of parameters drops from 8 (i.e. 2^n, where n is the number of parents) to 3. This is not as drastic a drop as the NoisyMAX distribution above, although as the number of parents increases, the NoisyOR distribution becomes more beneficial. However, increasing the number of parents also increases the chance that one of them violates the NoisyOR requirements.

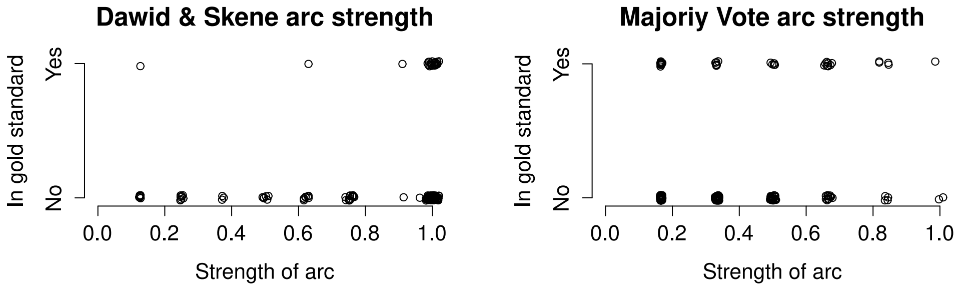
5.7. Limitations and Post-hoc Exploratory Analysis

Given the results in this chapter, it seems that the participants chosen were not an ideal proxy for domain experts. This section investigates whether there was still useful information to be gained from having a population of lay people with experience driving contribute to a BN for car insurance risk assessment. This type of analysis has its own limitations, as it is likely to fall victim to confirmation bias (Nickerson, 1998). The goal is not to attempt to formulate any hypothesis based on the available data after the experiment has been completed (Wagenmakers et al., 2012), and thus conclusions drawn should not be treated as evidence that lay people are a suitable substitute for experts in this type of experiment. Rather, the goal is to perform an exploratory investigation into the results to help guide future researchers construct appropriate experimental evaluations to determine more thoroughly if this is a worthwhile direction to take future research.

Specifically, this section investigated the questions which were answered the most emphatically during the survey. This was used to identify whether such emphatically answered questions may provide insight into how SEBN may be used in the future. Perhaps there are certain portions of BN structures that could be effectively elicited using this approach, leaving the remaining, more difficult parts to be elicited using traditional KEBN. Alternatively, perhaps a prototype BN can be quickly and cheaply developed via SEBN before investing a greater amount of time and cost on a lengthy traditional KEBN project.

5.7.1. The Cost of Excluding Arcs With Low Strength

Figure 5.25 shows the arcs included in the DS and Maj networks respectively. Each arc is plotted according to its strength (as calculated in Figure 5.7, p120) and whether it is a true positive, i.e. whether it is present in the *Gold* network. With respect to the DS arcs in Figure 5.25a, if all arcs with a strength of < 0.8 were to be excluded, regardless of whether they are in the *Gold* network (simulating the experience of not having a gold standard in the first place), then a total of 47 arcs are removed and 93 are retained. Of these 47 removed arcs, only two of them



(a) If arcs of strength ≤ 0.8 were to be excluded, then it would reduce the false positives by a large amount, while only excluding two true positives. A strength of 1 in this case represents arcs which were only included despite the *DS* prior only being was 0.1%, and a strength of 0 represents arcs that are included only when the prior is as high as 35%.

(b) The strength of arcs in the *Maj* networks give zero information about whether the arc should be in the *Gold* network or not. A strength of 1 represents arcs that are only included when 6 participants voted for them, whereas a strength of 0 represents zero people voting for an arc.

Figure 5.25.: Each point is an arc in the *Eval* network, that is either in the gold standard (true positive) or not (false positive).

are from the gold standard, whereas the remaining 45 are not. This indicates that the questions which were answered with an overwhelming level of agreement and subsequently included in the *DS* network may prove to be a helpful metric for when no gold standard is present.

The *Maj* arcs shown in Figure 5.25b tell a subtly different story, although it is difficult to discern visually. If arcs with a strength of < 0.6 were to be excluded, the results go from a total of 227 incorrect vs 36 correct to a lowly 18 incorrect but 13 correct. The ratio of incorrect to correct improves much more than the *DS* case, but for a cost of excluding half of the 36 true positive arcs.

Given this post-hoc analysis, when only the responses with particularly high agreement are taken into account, it seems that the lay participants with experience driving used in this evaluation are able to come up with good quality responses. It is suggested that future research should perform similar evaluations, perhaps also with a lay audience, but with some minor changes. Specifically, the researcher should outline before the experiment whether they value the inclusion of additional arcs at the expense of more noise in the resulting network, or the exclusion of incorrect arcs at the expense of less relationships in the network overall. If the former, than *DS* collation should be chosen. If the latter, than *Maj* empirically performed better during this evaluation. Subsequent evaluations could help to determine what the cutoff strength is for *Maj* and *DS* respectively in order to exclude poor quality responses and include good quality arcs in the resulting BN. The following section provides further post-hoc analysis of the arcs with high strength, in order to see if other patterns can be identified to help with future research into SEBN.

5.7.2. Investigation of Arcs With High Strength

The main assumption guiding the choice of lay people for use in this evaluation was that many of the concepts in the car insurance network would be known by lay people with experience driving, regardless of whether they have worked in the insurance field. To explore this assumption, the following two sections look at a selection of true and false positives from the evaluation to investigate whether the assumption turned out to be reasonable. The evaluation network arcs with a strength of $\geq 4/6$ (Table 5.5) are explored below by reviewing the questions that were asked in the survey in order to produce the them.

True positives with strong agreement This section highlights two questions from the true positive arcs with highest agreement in the evaluation (Table 5.4a).

> "Does the age of the clients vehicle direct [sic][11] influence any of these?
> - Airbags (Whether or not there are any airbags installed in the clients car)"

This had the highest level of agreement. This is unsurprising, given that air bags did not exist in vehicles beyond a certain date. Additionally, they have become more prevalent in new cars as time has progressed.

> "Does the quality of clients driving directly influence any of these? - Accident (Whether or not the client will be involved in an accident)"

[11] This was an error in the evaluation survey, it should've used the word "directly" instead of "direct".

(a) Correct arcs with high agreement.

From	To	Strength
VehicleYear	Airbag	1
DrivQuality	Accident	5/6
DrivingSkill	DrivHist	5/6
RiskAversion	DrivHist	5/6
RiskAversion	SeniorTrain	5/6
MakeModel	Airbag	4/6
SocioEcon	AntiTheft	4/6
Accident	ILiCost	4/6
SocioEcon	MakeModel	4/6
MakeModel	RuggedAuto	4/6
AntiTheft	Theft	4/6
HomeBase	Theft	4/6
RuggedAuto	ThisCarDam	4/6

(b) Incorrect arcs with high agreement.

From	To	Strength	Implied in *Gold* (indirectly)
DrivingSkill	Accident	1	*(DrivingSkill → DrivQuality → Accident)*
RiskAversion	Airbag	5/6	*(RiskAversion → MakeModel → Airbag)*
MakeModel	Theft	5/6	*(MakeModel → CarValue → Theft)*
Age	Accident	4/6	(*Age* is the top of the DAG, thus it indirectly influences all variables to some extent)
VehicleYear	Accident	4/6	*(VehicleYear → Antilock → Accident)*
Age	Airbag	4/6	(as above)
SocioEcon	Airbag	4/6	*(SocioEcon → MakeModel → Airbag)*
Age	MakeModel	4/6	*(Age→ SocioEcon → MakeModel)*
VehicleYear	Theft	4/6	*(VehicleYear → CarValue → Theft)*
DrivingSkill	ThisCarDam	4/6	*(DrivingSkill → DrivQuality → Accident → ThisCarDam)*
RiskAversion	ThisCarDam	4/6	*(RiskAversion → VehicleYear → RuggedAuto → ThisCarDam)*
MakeModel	AntiTheft	1	No
VehicleYear	AntiTheft	5/6	No
DrivQuality	DrivHist	5/6	No
ThisCarCost	Age	4/6	No
DrivHist	ILiCost	4/6	No
Accident	SeniorTrain	4/6	No
DrivingSkill	SeniorTrain	4/6	No

Table 5.5.: Arcs with high level of agreement arising from the evaluation study in Chapter 5.

It seems reasonable that a driver who is better at driving would also be better able to avoid accidents than those who are worse at driving.

The level of obviousness exhibited by the two questions above is indicative of all the other true positives with strong agreement in Table 5.4a. This is not surprising, given the post-hoc nature of the analysis undertaken. Naturally questions which were selected because they happened to be answered correctly will seem obvious. Nevertheless, it helps to contrast these with the false positives with strong agreement shown in the following section.

False positives with strong agreement This section highlights the false positives from the survey which have high agreement (Table 5.4b). Some of the false positives can be forgiven because they *do* represent causal relationships as seen in the gold standard, only with other mediating variables between them. This section will only analyse the false positives from Table 5.4b which are not represented as direct or indirect causal relationships in the *Gold* network. Each will fall into one of three error categories to better understand whether the incorrect responses are due to the choice of lay participants or other reasons:

Type A Question was worded correctly, but the answer is incorrect (evidence that lay people with experience driving are not a good supplement for insurance experts)

Type B Question was answered incorrectly, but perhaps the gold standard was actually incorrect (evidence that the choice of gold standard was poor)

Type C Question should have been relatively obvious, but were poorly worded (mistake on behalf of the researcher)

The false positives with the highest agreement were:

> "Does the type of the clients car directly influence any of these? - Anti theft device installed"

Perhaps a *Type B* error, as an argument could be made that sports cars or luxury cars would be more likely to have anti theft devices installed than family cars, due to their increased cost.

> "Does the age of the clients vehicle directly influence any of these? - Anti theft device installed"

If anti theft devices have become more common in newer generation cars, then this is a *Type B* error. Alternatively, an insurance expert may be able to produce evidence that the prevalence of such devices has stayed constant over time. In that case, this would be a *Type A* error.

> "Does the quality of clients driving directly influence any of these? - Driver history (If the client has a history of insurance claims)"

This seems like a reasonable causal relationship, and such could be classified as a *Type B* error. The reason it was not encoded in the *Gold* network was likely because the Binder et al. (1997) article is investigating latent variables which results in a relationship from *DrivQuality* ← *DrivingSkill* → *DrivHist*, where *DrivingSkill* is the latent variable that explains the relationship between *DrivQuality* and *DrivHist*. As such there is indeed a relationship between *DrivQuality* and *DrivHist* in the *Gold* network, just not a causal relationship.

> "Does the cost to the insurer to fix clients car directly influence any of these? - Age of the client"

This is perhaps the most confusing response given by survey participants as it has resulted in responses that violate causality. It is clear that the cost of fixing a car does not influence anyone's age. The wording of the question does not seem particularly ambiguous either, indicating that this is a *Type A* error.

> "Does the clients driving skill directly influence any of these? - Advanced driver training (Whether the client has undergone additional training after obtaining their license. Some companies may refer to this as "Skilled Driving" or "Defensive Driving" courses)"

This question seems to have been interpreted in an inverse manner. It seems likely that advanced driver training would influence a drivers skill, but not the reverse. As such, this is a *Type A* error.

> "Does the clients driving history directly influence any of these? - Cost to insurer for liability/property (The total cost to the insurer for 3rd party property damage, due to an accident caused by the client)"

This appears to be a *Type C* error, due to the confusion between the persons past history of insurance claims with previous insurers, and a potential future claim with a new insurer. If this was articulated more clearly, then perhaps it would not have caused confusion. Alternatively it is possible that participants though that past incidents are indicative of future incidents, which may make this a *Type B* error.

> "If the client becomes involved in an accident, will it directly influence any of these? - Advanced driver training (Whether the client has undergone additional training after obtaining their license. Some companies may refer to this as "Skilled Driving" or "Defensive Driving" courses)"

This is likely a mix of *Type A and C* error. It is *Type C* because there are ambiguities in the wording whereby the chance of an accident is a potential future event, whereas the advanced driver training is something the client has already undertaken before signing up for insurance. If this was misinterpreted as the chance that a driver will undertake advanced training in the future, then it becomes a *Type A* error. This is due to the knowledge an expert would have about whether insurers encourage people to undertake driver training in response to having an accident or not.

Given the post-hoc analysis above, it seems that indeed there was some *Type A* errors which would not be present had a cohort of insurance experts been enlisted for the evaluation survey. However there are also many other errors which are potentially not related to the choice of participants. These *Type B* and *C* errors are down to other factors that could be addressed with future research into SEBN.

5.7.3. Summary of Post-hoc Analysis

This post-hoc analysis has shown that many of the false positive arcs with high agreement were valuable inclusions in the resulting BN structure. In particular, they tended to indeed be causal relationships that are worthy of consideration. The problem arises when there is mediating variables that would better encode the causal relationship. As such, it is suggested that future research focus on the indirect vs direct anomalies explained in Section 4.6.2 (p95) to make sure that such relationships in the evaluation are able to be identified.

Additionally, at the culmination of any future evaluations, arcs with a high level of agreement should be selected for inclusion in the resulting network. Given the post-hoc empirical observations outlined above, a *DS* prior of between 5% and 10% was a good choice, but this would require further experimentation before deciding on a usable prior. Arcs with a lower strength should then be further investigated in a traditional KEBN session with experts. This allows for at least part of the BN to be elicited at a lower cost and a simpler manner using SEBN, reducing the burden on experts when it does come time to perform traditional KEBN to complete the model.

5.8. Chapter Summary

This chapter documented the evaluation of SEBN which took place using the BNE software to elicit the structure of a BN for car insurance risk assessment. The goal was to conduct an online survey and collate the responses into a BN structure, then compare to a known gold standard and measure how close the elicited structure was. Section 5.2 outlined the different network structures that were created in order to evaluate the effectiveness of EM vs Majority Vote collation.

A good result would have been where the elicited structures were close to the original BN in terms of SHD, F1 scores, and BIC scores. The process also should have taken less time of each participant compared to if they were asked to partake in one or more face to face interviews.

The results from Section 5.3 show that the SHD, F1, and BIC compared unfavourably to the benchmark network. Section 5.3.5 summarised the results of each scoring metric, and showed that Majority Vote produced the individual networks with the best scores. However the EM algorithm is more predictable, in that a lower prior almost always produces better results. Although the Majority Vote also trends towards better scores with an increased threshold, there is a point at which it quickly deteriorates. This point may not be apparent in the absence of a gold standard. The EM algorithm was more robust during the post-hoc analysis which artificially reduced the number of responses available for collation (Section 5.3.4). This indicates it may be better able to deal with a lower number of participants. Further research is warranted to investigate whether the predictability of EM vs the sensitivity of Majority Vote is exhibited when other BN structures are elicited using SEBN. Also, the artificial reduction in the number of responses should be investigated by eliciting the same network multiple times with a different number of experts.

As for the time spent by participants when eliciting the structure, the semi-structured nature of traditional interviews makes it hard to compare to SEBN, given the scope of this thesis. Nevertheless, Section 5.4 showed that the median amount of time spent by participants using BNE was under 30 minutes and the most was below 1 hour, which is less than many knowledge elicitation sessions. This is an improvement even when taking into consideration structure elicitation accounts for perhaps $\frac{1}{4}$ of the entire BN elicitation. Future research should conduct both SEBN and traditional KEBN for the same BN for even more comparable metrics.

The following chapter elaborates on SEBN, by extending to facilitate the elicitation CPTs in addition to BN structures.

6. Calculating BN Probabilities Through Survey Based Elicitation

This chapter introduces the second part of SEBN, used to elicit the CPTs of a BN structure via expert elicitation using online surveys. More specifically, it adopts ideas from past research including Das (2004); van der Gaag et al. (1999) and Saaty (1990), discussed in the literature review in Section 2.2.5 (p36) and shows how such approaches can be used for eliciting probabilities for BNs in an efficient manner. Any system for reducing the magnitude of the CPT elicitation task by utilising online surveys will likely have to incorporate many of the techniques from Section 2.2.5, choosing the most appropriate one for each CPT requiring elicitation.

Many of the ideas in this chapter are similar to those presented in Chapter 4 when discussing the elicitation of BN *structures* via SEBN. Where relevant, the discussion here will delegate to the previous writing to reduce repetition.

6.1. Overview

This section shows a very general overview of the process, to be elaborated on in subsequent sections as each step is explained in detail. The broad overview is shown in Figure 6.1. The reader can skip forward to Figure 6.10 (p164) to view the final process in its entirety.

Generate Questions Section 6.2 discusses how the various questions will be generated. It will show how the questions will differ based on the local structure of any given variable. For example, nodes with zero parents result in a different set

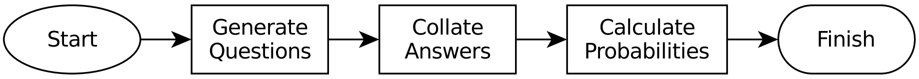


Figure 6.1.: Simplified flowchart showing the process for eliciting probabilities with surveys. It does not show anything particularly revolutionary, but will be expanded on in greater detail throughout this chapter.

of questions to those with a single parent, which are different from those with multiple parents.

Allocate Questions The allocation of questions to experts (Section 6.3) is very similar to Section 4.4 (p85), which discusses reducing the burden on experts by only asking a subset of questions. It is different in this chapter in that the specific types of questions to be allocated will differ, and the total number of questions may not be known.

Collate Answers Section 6.4 shows two methods in which probabilities can be collated together from multiple experts in order to produce an authoritative set of CPTs for the final BN. The first approach shown is to take the average of all estimations for a given probability and then normalize. The second is to make use of the EM algorithm to take into account the differences in how each expert answers the questions allocated to them.

6.2. Generate Questions

The method prescribed in this chapter considers three distinct types of BN node, and formulates different sets of questions based on this. The first two cases (zero or one parent variables) focus their efforts on reducing the cognitive load of probability elicitation, rather than reducing the number of probabilities to be elicited. The third makes use of the weighted sum algorithm (Das, 2004) discussed in the literature review in Section 2.2.5 (p36) to ask about only a subset of all probabilities, then infer the remaining probabilities.

The simplest case is when a node has no parents, in which case the expert is required to answer one question for each state of that variable, about its marginal probability. In the case where there is a single parent, the expert is asked to elicit a conditional probability for each of the variables states, conditioned on each state the parent can take. Finally, when a node has multiple parents, experts are first asked for "Compatible Parent Configurations" (CPC) as described by Das (2004), then the conditional probability of each child variable state, conditioned on the CPCs, followed by questions to ascertain which parent has the greatest influence on the child variable. This information is then processed according to the weighted sum algorithm to fill in the remainder of the CPT.

A Word on the Use of Flow Chart Syntax Wherever possible, this thesis attempts to make use of standardised syntax for flowcharts (ISO 5807:1985). The only exception is in this chapter, where nested loops are used heavily. The usual way to encode a loop in a flowchart is a branch statement which checks for loop termination, and returns to the beginning if the loop is to continue. This can quickly become unwieldy and hard to interpret when nested loops are involved. This chapter uses a non-standard syntax that represents loops as plates containing flowchart elements, which can be nested (e.g. Figure 6.3 shows two nested plates: "For each variable", and "For each state"). It is hoped that this is a more concise syntax which makes it easier to interpret and understand the diagrams.

6.2.1. Questions Required for Variables With no Parents

The simplest situation is when a particular variable has no parents. In this case, the questions needn't be concerned with conditional probabilities at all. Additionally, the number of questions required depends only on the number of states present in the variable. This is distinct from conditional probabilities, where the number of parameters grows exponentially with the number of parents. This section will use the BN node shown in Figure 6.2 as an example to illustrate how the questions are generated. Figure 6.3 illustrates the process involved in generating questions here.

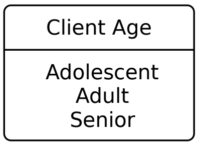


Figure 6.2.: Single variable with no parents, including the three possible states the variable can take.

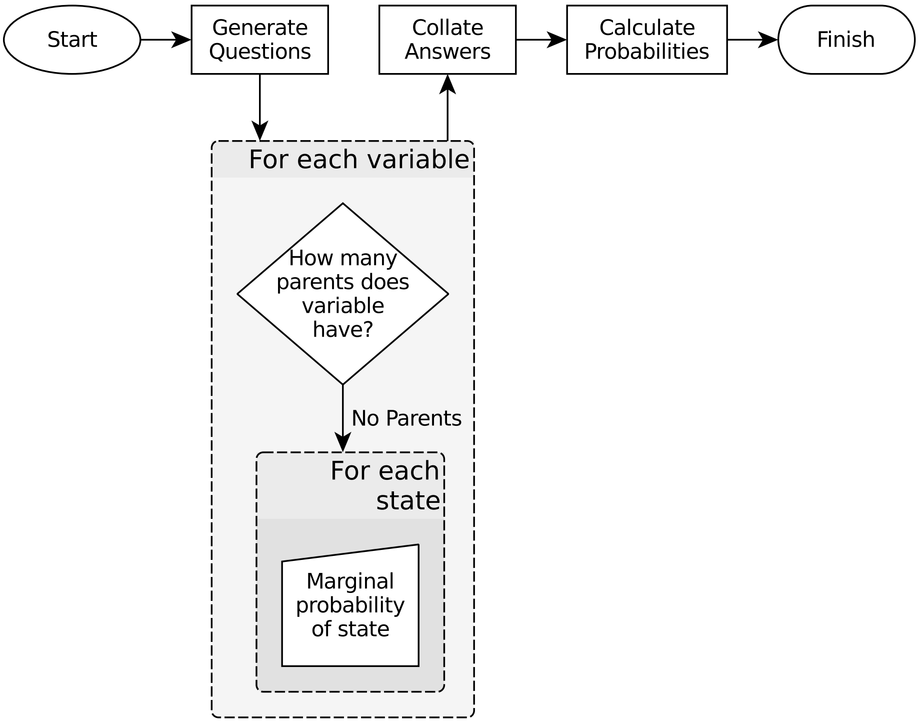


Figure 6.3.: Elicit the marginal probability of every state for each variable.

The questions to elicit the marginal probability of each state of the variables are formulated in the following way:

> What is the likelihood of the following scenario?
>
> `State`

Where `State` is an English phrase manually provided by the knowledge engineer, describing the state of the variable with whatever context is required to make it coherent and understandable by experts. There may be an urge to combine the variable name and the state name programatically, but this is discouraged. Hope et al. (2002) propose the following format (for nodes with multiple parents):

> "Consider that `Parent1` is `State1` and that `Parent2` is `State5` . What is the chance that `Child` is `State3` ?" (Hope et al., 2002, p7)

However, this will result in mechanical sounding and jarring phrases. For example, consider the variable *Client Age* and one of its states *Young adult*, where the `State` is manually entered as "Client is a young adult" by the knowledge engineer:

- How likely is the following scenario: Client Age is Young Adult?
- How likely is the following scenario: Client is a young adult?

While the former is constructed in a similar fashion to Hope et al. (2002), the latter "Client is a young adult" is preferable, even though the variable name *Client Age* isn't present in the sentence. Appendix F (p269) presents a more comprehensive set of examples for generating questions of this and every other type.

The response to questions of the form "How likely is the following scenario: `State`?" should be a probability value between zero and one. In order to ease the cognitive burden of eliciting these probability values, the question should be asked by presenting the scale proposed by van der Gaag et al. (1999) and advocated by Korb and Nicholson (2011, p327). The scale in Figure 6.4 shows the seven different options which are shown to experts, and an English label of what each item on the scale represents.

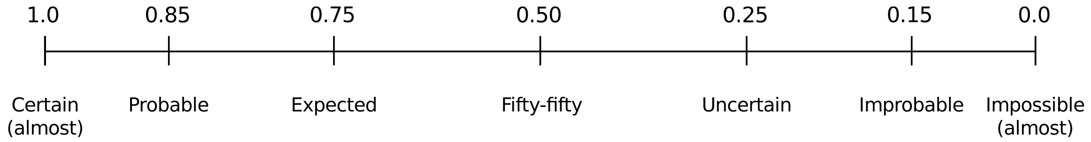
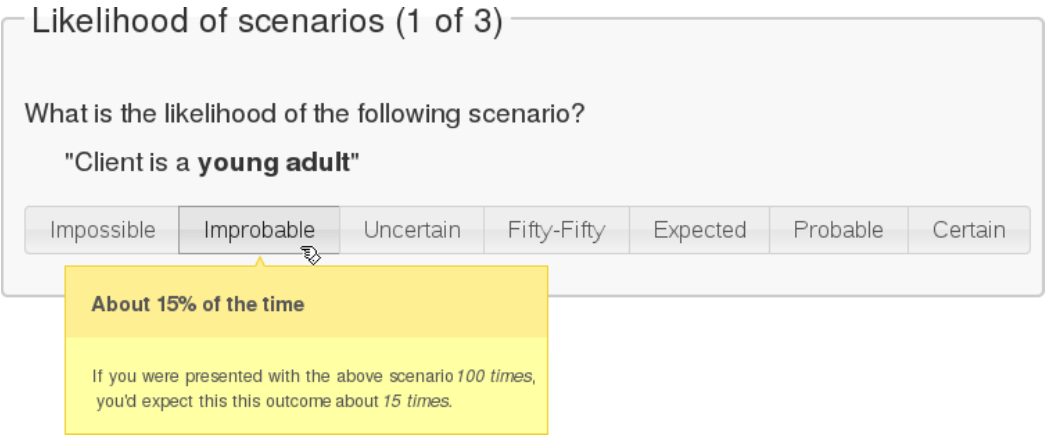


Figure 6.4.: "The Fragment of Text and Probability Scale for the Assessment of the Conditional Probability" adapted from van der Gaag et al. (1999).

The BNE software uses the van der Gaag et al. (1999) scale, and augments it by providing contextual help which utilises relative frequency elicitations (Price, 1998) rather than probability elicitations (Figure 6.5). Price (1998) showed probabilities elicited from questions framed using relatively frequencies "tended to be lower, exhibit less scatter, and express complete certainty less often than judgments in response to the probability elicitation question".



Likelihood of scenarios (1 of 3)

What is the likelihood of the following scenario?

"Client is a **young adult**"

About 15% of the time

If you were presented with the above scenario *100 times*, you'd expect this this outcome about *15 times*.

Figure 6.5.: User interface of BNE showing a question for the *Client Age* variable, which has zero parents in the insurance network. Whenever a button is hovered over, the tool tip is exposed to clarify the meaning of the words from the van der Gaag et al. (1999) scale.

6.2.2. Questions Required for Variables With a Single Parent

This section will use the example child node *Liability Cost* and its single parent node *Accident* (Figure 6.6). All conditional probabilities for the child node are elicited using the same scale depicted in Section 6.2.1. The only difference is that the probability being elicited is a *conditional* probability, conditioned on each of the states of the parent variable. Figure 6.7 augments the flowchart to include the single parent case.

The first part of the resulting questions are similar to those from the case where the node has no parents (Section 6.2.1). This is followed by a description of the parent state which the child state is being conditioned on:

> What is the likelihood of the following scenario?
>
> `ChildState`
>
> If we know that:
>
> `ParentState`

Figure 6.8 shows how the questions look in the BNE software, incorporating the states of the parent variable. Appendix F (p269) provides further examples of questions generated in this fashion for the insurance network.

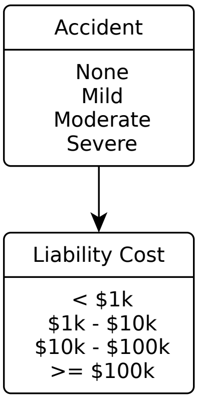


Figure 6.6.: Child variable with a single parent. The states the variable can take are shown also.

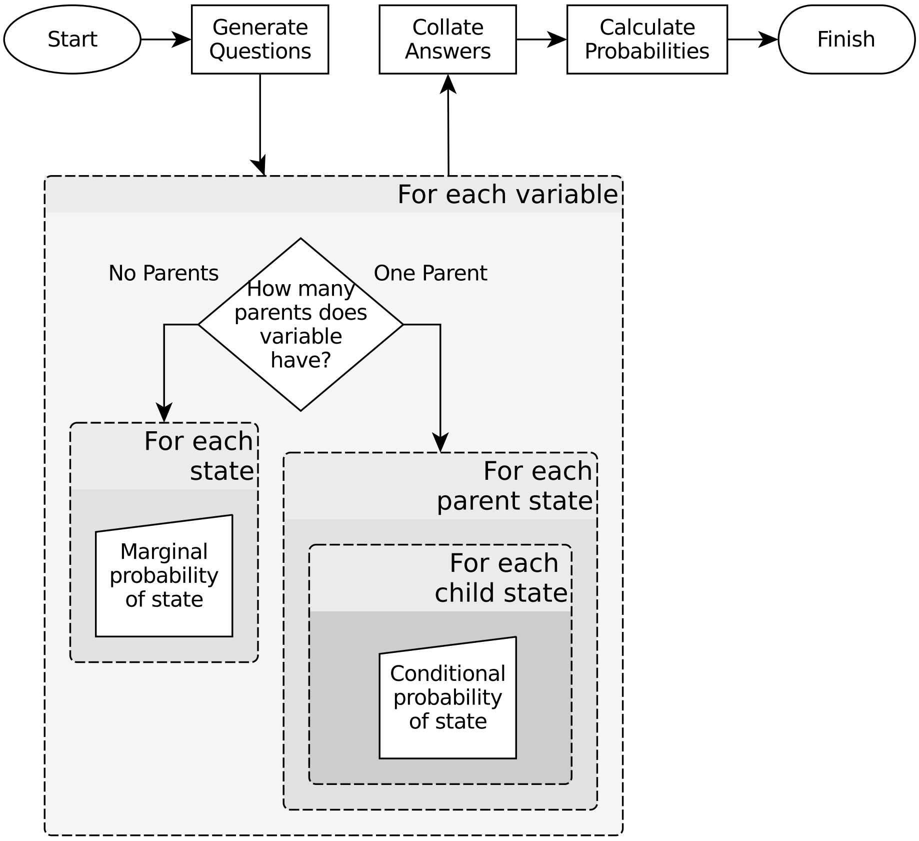
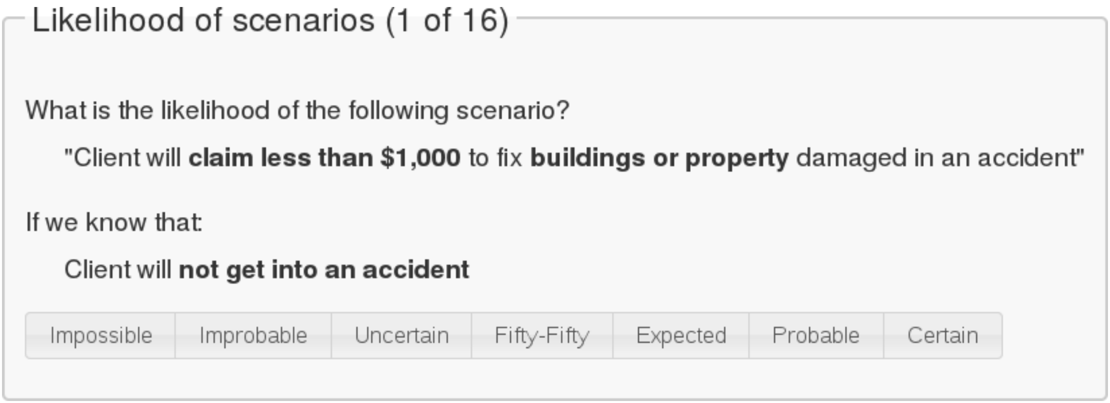


Figure 6.7.: Elicit the marginal probability of every state for each variable.



Likelihood of scenarios (1 of 16)

What is the likelihood of the following scenario?

"Client will **claim less than $1,000** to fix **buildings or property** damaged in an accident"

If we know that:

Client will **not get into an accident**

Figure 6.8.: Screenshot of BNE when asking about the *Liability Cost* variable, which has a single parent of *Accident*.

6.2.3. Questions Required for Variables With Multiple Parents

Figure 6.10 elaborates on the flowchart to include the process for eliciting CPTs of variables with multiple parents. Whereas the previous sections documented the elicitation of an *entire CPT*, this section proposes questions that directly elicit only a subset of probabilities for each CPT. The weighted sum algorithm is then used to fill in remaining CPT entries. The end result is that a reduced number of questions need to be asked from the experts, compared to if they were to populate the entire CPT. This involves three distinct phases, to be discussed below:

- Eliciting compatible parent configurations (CPCs).
- Eliciting conditional probabilities.
- Eliciting relative weights of each parent.

Throughout this section, the child node *Car Value* and its three parents *Mileage*, *Vehicle Year*, and *Car Type*, will be used as the example (Figure 6.9).

Compatible Parent Configurations (CPC) The first thing to do when there are multiple parents is to elicit the relevant CPCs. The process is explained in Algorithm 6.1, whereby each parent variable is iterated over, and for each state of that variable, the most likely state of all other parents is elicited.

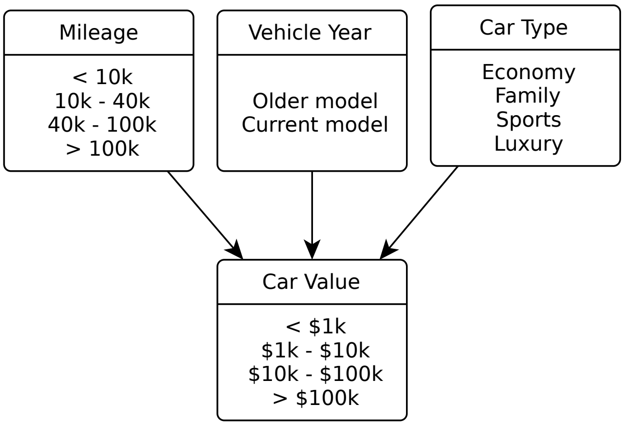


Figure 6.9.: Single variable with multiple parents. The states that each parent and the child variable can take are shown also.

Algorithm 6.1 Eliciting compatible parent configurations for a particular child variable.

```
$\mathbf{P} \leftarrow$ Parent nodes
$\mathbf{Q} \leftarrow \varnothing$                                        ▷ Questions

for all $\mathbf{P}$ as $\mathbf{p_x}$ do
    for all STATES($\mathbf{p_x}$) as $\mathbf{s_{xi}}$ do
        $\mathbf{P_{others}} \leftarrow \mathbf{P} \setminus \{\mathbf{p_x}\}$
        $\mathbf{Q} \leftarrow \mathbf{Q} \cup \{$ "When $\mathbf{p_x}$ is in state $\mathbf{s_{xi}}$, what states would you expect $\mathbf{P_{others}}$ to be in?"$\}$
    end for
end for
```

Figure 6.11 shows the questions to be asked in order to elicit the CPCs for the parents of the *Car Value* variable (Figure 6.9). Once the expert has selected the two most compatible parent states for the *Vehicle Year* and *Car Type* variables, then questions are repeated, for the next state of the mileage variable:

> "If *Client's car has driven between 10,000 and 40,000kms*, then I expect..."

This is repeated until all of the states of the *Mileage* parent have been exhausted. Then, the same process is conducted for the *Vehicle Year* variable, whereby the most compatible states of *Car Type* and *Mileage* are elicited. Finally, the same is done for the *Car Type* parent, resulting in one CPC for each state of each parent.

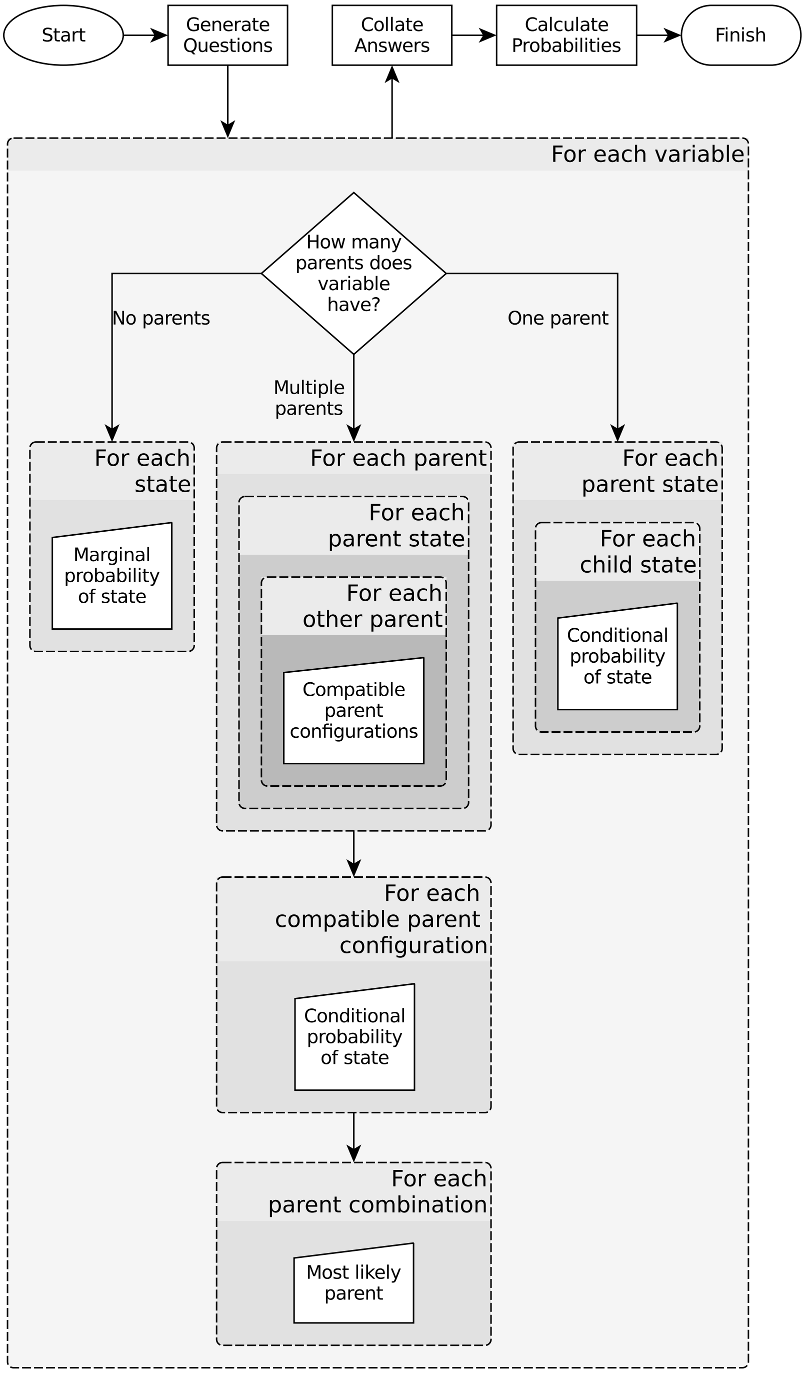
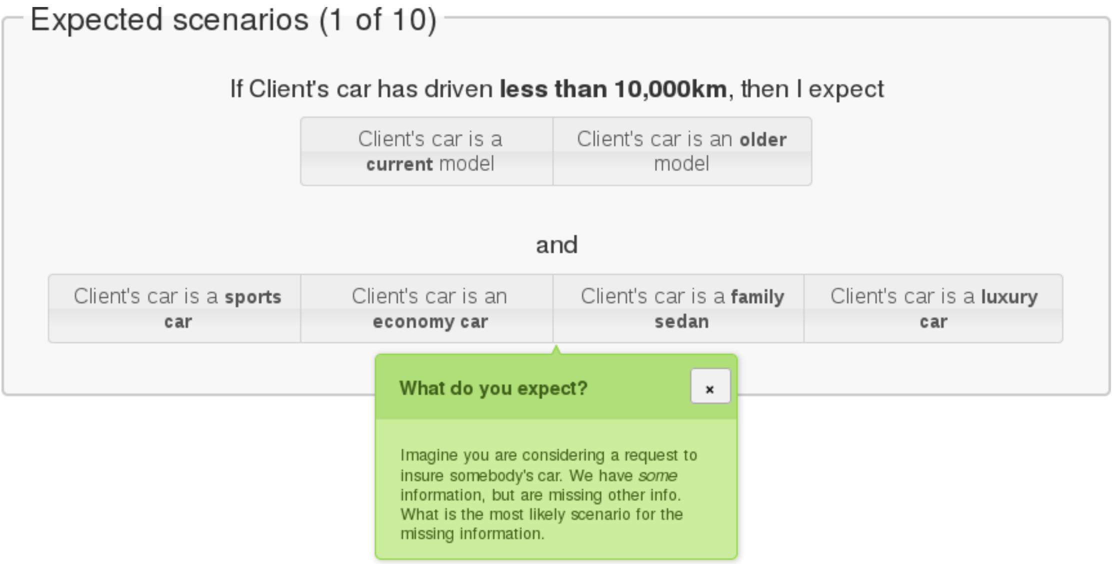


Figure 6.10.: The process after adding the elicitation procedure for variables with multiple parents. The flowchart for multiple parents includes questions for both the weighted sum algorithm and also the AHP algorithm.



Expected scenarios (1 of 10)

If Client's car has driven **less than 10,000km**, then I expect

Client's car is a **current** model
Client's car is an **older** model

and

Client's car is a **sports car**
Client's car is an **economy car**
Client's car is a **family sedan**
Client's car is a **luxury car**

What do you expect?

Imagine you are considering a request to insure somebody's car. We have *some* information, but are missing other info. What is the most likely scenario for the missing information.

Figure 6.11.: Eliciting the CPC from the three parents of *Car Value* shown in Figure 6.9: *Mileage*, *Vehicle Year* and *Car Type*.

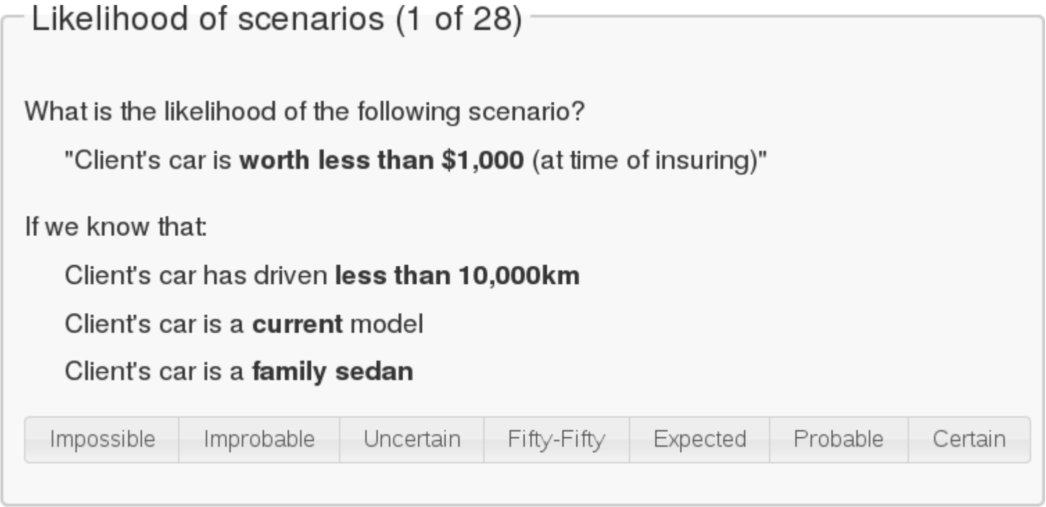
Conditional Probability Once CPCs have been elicited for each parent state, the conditional probability of each child state, conditioned on the CPCs needs to be elicited. These questions are posed the same way as for the single parent case, using the van der Gaag et al. (1999) scale (Figure 6.4, p159). The difference is that instead of conditioning on each of the states of the single parent variable, each child state is conditioned on each of the CPCs elicited from the previous questions.

For example, if

$$CPC\left(Mileage =< 10k\right) =$$
$$\left\{Mileage =< 10k, Vehicle\,Age = current, Make\,Model = family\,sedan\right\}$$

Then there will be four questions of the format shown in Figure 6.12, one for each of:

- $\Pr\left(Car\,Value < \$1,000 | CPC\left(Mileage < 10k\right)\right)$
- $\Pr\left(Car\,Value < \$10,000 | CPC\left(Mileage < 10k\right)\right)$
- $\Pr\left(Car\,Value < \$100,000 | CPC\left(Mileage < 10k\right)\right)$



Likelihood of scenarios (1 of 28)

What is the likelihood of the following scenario?

"Client's car is **worth less than $1,000** (at time of insuring)"

If we know that:

Client's car has driven **less than 10,000km**

Client's car is a **current** model

Client's car is a **family sedan**

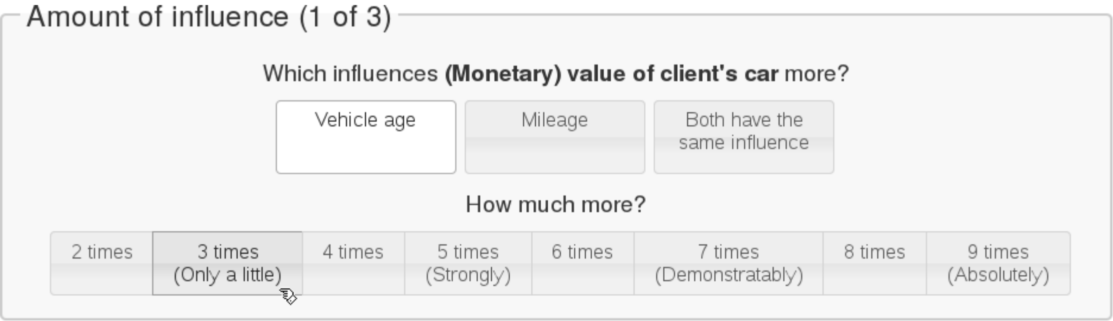
Figure 6.12.: Question to elicit the $\Pr(Car\,Value < \$1,000|$ $Mileage < 10k, Vehicle\,Age = current, Make\,Model = family\,sedan)$, where the three parent states represent the CPC when *Mileage = <10k*.

- $\Pr\left(Car\,Value \geq \$100,000|CPC\left(Mileage < 10k\right)\right)$

Relative Weight of Each Parent After eliciting conditional probabilities, the final step in the weighted sum algorithm is to elicit weightings for each parent variable. That is, which variable has the largest influence on the state of the child variable. The original algorithm asks for a weight for each parent directly from experts, such that they all sum to 1 (Das, 2004, p10). However, other researchers have successfully made use of pairwise comparisons as a more intuitive way to elicit relative weightings between multiple choices (e.g. Chin et al., 2009). Thus, SEBN makes use of the Analytic Hierarchy Process (AHP, Saaty (1990)) to elicit these weightings in a manner more fitting of a survey. That is, instead of explicitly asking the experts for numeric weights, it asks them to perform pairwise comparisons of each parent and then use the results of those comparisons to calculate weightings.

A pairwise comparison of each parent is shown to the participant, and they state whether one of them is more influential, or they are equally influential (Figure 6.13).

Once the expert has mentioned which variable they deem to be the most influential, then they are asked how much more influential that parent is. This is done



Amount of influence (1 of 3)

Which influences **(Monetary) value of client's car** more?

How much more?

Figure 6.13.: Questions required for a pairwise comparison between *Vehicle Age* and *Mileage*. The responses are used as input into the AHP process to determine relative weightings between all parent variables of *Car Value.*

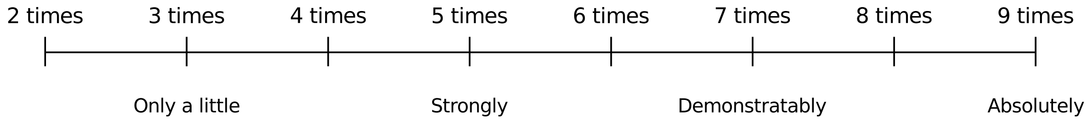


Figure 6.14.: The scale used to elicit the relative difference in influence that two different parents have on a child variable. The numbers adapted from Saaty (1990, p112) and the labels from Monti and Carenini (2000, p503).

by presenting them with options from 2 times more influential to 9 times more influential. The numbers were derived from the AHP process documented in Saaty (1990, p112). To make them more understandable, the labels used in SEBN for values between 2-9 are from Monti and Carenini (2000, p503).

Note on Increased Number of Questions Due to Using AHP

SEBN is about reducing the burden on experts. Chapter 4 addresses this almost exclusively by reducing the number of questions each participant needs to answer. This chapter, and specifically the choice to adopt AHP in place of directly eliciting relative weights, is different. The goal here is to reduce the cognitive burden by decomposing the problem of eliciting weights into more manageable questions in a more natural format. In fact, the number of questions is increased somewhat significantly when moving from “Assign relative weights to each parent to indicate the relative strength of influence on the child node” to “pairwise comparisons of each of the parent variables”. The original approach needs only n questions, whereas the AHP requires between $\frac{n^2-n}{2}$ and $n^2 - n$ questions. It is anticipated that, given most BN structures try to minimise the number of parent nodes in BNs, this increase in questions only becomes a problem at higher values of n. This is not much of a problem in the context of BNs, as it is often advocated to keep the value of n low for any given node to prevent combinatorial explosion (Neil et al., 2000, p16).

6.3. Allocating a Subset of Questions to Each Survey Participant

As with the structure elicitation in Chapter 4, only a subset of questions are allocated to each participant. This section will highlight the similarities and differences from the earlier discussion in Section 4.4 (p85).

6.3.1. What is Allocated

Section 6.2 discussed the specific questions which are generated for variables with zero, one or many parents. Variables with many parents produce more questions, and the specific questions for eliciting conditional probabilities may vary, depending on earlier responses to the questions eliciting CPCs. Given motivation for introducing CPCs is so that experts can reason in a coherent frame of thought (Das, 2004), it would be disadvantageous to take the CPCs from one expert and use them to ask about conditional probabilities from a different expert. Thus, individual variables should be allocated to experts, and then each participant is asked to answer all the relevant questions for variables allocated to them, according to the rules outlined in Section 6.2. For example, if an expert is allocated the *Age* variable which has zero parents, they will be asked multiple questions in order to elicit the marginal probability of that variable. Alternatively, if they were allocated the *CarValue* variable with its three parents, then they will need to first elicit the CPCs, then the conditional probabilities for each child state given each CPC, and finally the AHP questions in order to produce weightings for each parent.

The smallest block of questions that are required in order to elicit a full CPT for a given variable is defined by:

$$n_{var} = \begin{cases} s_{var} & p_{var} = 0 \\ s_{var} \times s_p & p_{var} = 1 \\ \left[CPC + CP + \frac{AHP}{2}, \; CPC + CP + AHP\right] & p_{var} > 1 \end{cases} \tag{6.1}$$

Where:

- n_{var} is the number of questions required to parameterise the CPT for variable var
- p_{var} is the number of parents for variable var
- s_{var} is the number of states the child variable can take
- s_p is the number of states of a single parent
- $CPC = s_{var} \times CP$
- $CP = \sum\limits_{i=1}^{p_{var}} s_{pi}$
- s_{pi} is the number of states parent i can take
- $AHP = p_{var}^2 - p_{var}$

Note that in the $p_{var} > 1$ case, the number of questions is a range. This is due to the situation in which an expert responds to a pairwise comparison by saying that they are equally influential. In such a case, there is no need to ask a subsequent question about how much more influential either parent is. In the pathological case, only half the maximum number of AHP related questions would be asked if all parents were to be judged as equally influential.

6.3.2. Question Allocation Strategies

The way in which different variables can be allocated to various experts is discussed in detail in Section 4.4 (p85). The same strategies apply to CPT elicitation in SEBN, whereby experts could be allocated variables based on the class the variables belong to, facilitating specialist experts to provide answers to targeted questions. As with the structure elicitation, the BNE software elects to allocate the least allocated questions to each expert as they sign up for the survey.

6.3.3. How Many Allocated Questions is Enough?

Outlined in Section 4.4.3 (p87) is four variables used when calculating the expected number of allocations for a given expert. These are:

1. How many questions are there to be answered in total (n_{tot})?
2. How often should like each question be answered by an expert (a)?
3. How many experts are available (n_e)?
4. How many questions are allocated to each expert (n_q)?

These are equally relevant when allocating variables for CPT elicitation, as are the resulting equations Eq. 4.1, Eq. 4.2, and Eq. 4.3 (p88). As discussed in Section 6.3.1, the smallest block of questions which should be allocated to a particular expert is defined by n_{var} in Eq. 6.1 (p169). Thus, $n_{tot} = \sum_{i=1}^{m} n_{var,i}$ where m is the number of variables in the BN, and $n_{var,i}$ is the value of n_{var} for variable i.

Other than these differences, the steps for deciding how many questions to allocate to each expert is the same as in Section 4.4.3 (p87). That is, the knowledge engineer needs to take into considerations information such as:

- Is the number of experts known before the survey begins?
- How much time is likely required to answer questions?
- What is the maximum amount of time that each expert is expected to afford?

Answering these questions will allow Eq. 4.1, Eq. 4.2 and Eq. 4.3 (p88) to be solved and the number of allocations to be decided.

6.4. Collating Answers

The data which will be collated into CPTs will be the various full CPTs elicited by experts, *after* the weighted sum algorithm has been applied to the raw survey responses. This is because different experts will potentially have been asked different questions depending on how they responded. Specifically, it is likely that different experts may elicit different CPCs, which in turn results in different conditional probabilities being requested. Thus, it is not possible to combine different participants raw responses together before processing into CPTs.

Mean The mean is perhaps the simplest mechanism for taking estimations from multiple experts for the same conditional probability and combining them into one conditional probability value. In order to do so, each expert who answered questions resulting in a particular CPT will have their values for each probability summed. This is then normalised so that the CPT only contains probabilities that sum to 1, for each combination of parent states.

Expectation Maximization Algorithm A more advanced mechanism for taking multiple estimations of the same value and combining them is the EM algorithm. This was discussed in Section 4.5 (p90) with respect to structure elicitation. As with the mean discussed above, the resulting probabilities will require normalization after applying the EM algorithm.

6.5. Trade-offs

This section will document the various trade-offs which have been identified in the processes of refining and documenting the system for using SEBN to elicit CPTs. The goal is to allow knowledge engineers to make the most informed decision when implementing this technique for BN elicitation, with full knowledge of the possible trade-offs which need to be considered.

6.5.1. Preventing Duplicate Questions When Duplicate CPCs Arise

It is quite likely experts will end up with some duplicate CPCs for differing parent states. In such cases, the conditional probability of the child state given that CPC need only be elicited once. For example, it may be the case that:

$$\begin{aligned} &CPC\left(Mileage < 10k\right) = CPC\left(Vehicle\,Age = current\right) = \\ &\quad \left\{Mileage < 10k, Vehicle\,Age = current, Make\,Model = family\,sedan\right\} \end{aligned}$$

In the above case, it would not be strictly necessary to elicit both of:

$$\begin{aligned} &\Pr\left(Car\,Value < \$1,000 | CPC\left(Mileage < 10k\right)\right) \\ &\qquad\qquad \Pr\left(Car\,Value < \$1,000 | CPC\left(Vehicle\,Age = current\right)\right) \end{aligned}$$

as they will both represent the same CPT in the weighted sum algorithm. Implementers of SEBN may choose to intentionally ask more than once, as a way to check how consistently experts respond to the question. Monti and Carenini (2000, p507) investigated inconsistencies in expert elicited probabilities. They found that forcing the expert to confront such inconsistencies didn't seem to have any specific benefits, given their expert would usually choose the midpoint between two inconsistent estimates. The BNE software opted to only ask for the conditional probability once.

6.5.2. Randomising question order

When presenting a series of questions to an expert, they can be either randomised or presented in order by iterating over all variables and their states. If the questions are presented in order, then people will have time to comprehend the questions better, and build a proper mental model of what it is asking. In their work on crowd sourcing, Organisciak et al. (2012) found that people who had time to familiarise themselves with the task at hand performed better at answering questions. However in the case of the probability elicitation questions in this chapter, two questions can look the same at first glance when in fact they are asking about two different conditional probabilities. Section F.2 (p270) shows several examples of questions which were generated during the evaluation in Chapter 7, highlighting the way in which some questions can appear similar. If the order is randomised, then each subsequent question should be sufficiently different from the previous such as to cause the expert to read it in full. The BNE software does *not* randomise the order of the questions

6.6. Chapter Summary

This chapter documented the series of steps required in order to use online surveys to elicit the CPTs of a BN. This chapter complements Chapter 4 (p73) which focussed on eliciting the structure of a BN. Put together, these two chapters form SEBN, prescribing a comprehensive system for using online surveys to elicit BNs. The following chapter documents the experimental evaluation to investigate the propositions documented in Section 3.3 (p60) and discuss whether SEBN is a suitable approach to use to elicit CPTs BNs.

7. Evaluating Probability Elicitation

The aspect of SEBN focussed on eliciting CPTs (described in Chapter 6) is evaluated in this chapter. The evaluation took the shape of an online survey using the BN Elicitator (BNE) software (Serwylo, 2013). In the language of Venable et al. (2012) it was primarily a naturalistic, ex post evaluation, as it was conducted in the same manner that a live survey to elicit the CPTs for a BN would be. The main difference is the choice of participants, discussed in Section 7.1.2.

This chapter will be organised as follows: Section 7.1 describes the method of evaluation. After describing the experimental setup, it analyses the participants who took part in the online survey. Results of the survey are discussed in Section 7.2 focusing on the CPTs elicited during the survey and how they compare to those in the gold standard network. Section 7.3 verifies how coherent the responses provided by the participants are, and Section 7.4 shows the amount of time spent by each participant while completing the survey. Following this, Section 7.5 investigates the assumptions made by Das (2004) when proposing the weighted sum algorithm. The assumptions are tested empirically by comparing the probabilities calculated using the algorithm with the corresponding probabilities which were explicitly elicited from participants. Finally, a summary of the evaluation is presented in Section 7.6.

7.1. Experimental Method Used for Evaluation

This section documents the experiment which took place in order to address the propositions laid out in Section 3.3 (p60). The experiment took the form of an online survey about car insurance. Section 7.1.1 describes the method in more

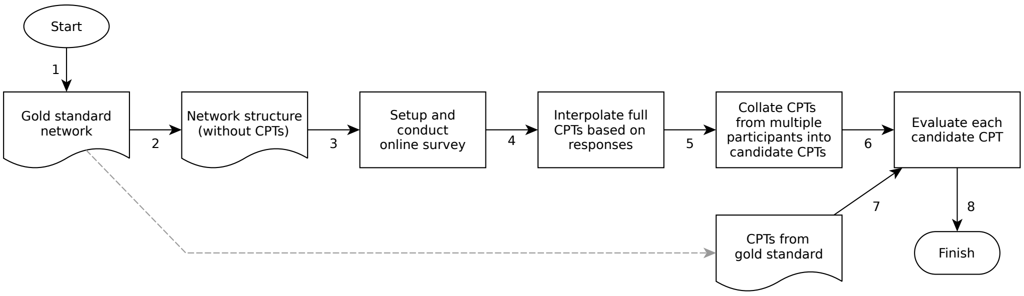

Start
1
Gold standard network
2
Network structure (without CPTs)
3
Setup and conduct online survey
4
Interpolate full CPTs based on responses
5
Collate CPTs from multiple participants into candidate CPTs
6
Evaluate each candidate CPT
7
CPTs from gold standard
8
Finish

Figure 7.1.: A broad overview of the process used to evaluate the elicitation of CPTs via SEBN.

detail, followed by a summary of the participants who took part in Section 7.1.2. The results of the experimental survey will be presented later in the chapter.

7.1.1. Overview of Experimental Method

The goal with this evaluation was to elicit the CPTs for a BN structure using SEBN (described in Chapter 6, p155). These elicited CPTs could then be compared to a known gold standard network to see how closely the elicited probability distributions match. Figure 7.1 shows a high level overview of the experiment which was undertaken.

The BN structure which was parameterised during this evaluation was that of the car insurance network (Binder et al., 1997) discussed in more detail in Section 5.1 (p104). Although the evaluation could have used the structure resulting form the previous evaluation of structure elicitation (Chapter 5), this would not have been an appropriate gold standard. It did not contain any CPTs at all, and thus the CPTs elicited during this evaluation would not be able to be compared to anything to verify if they were accurate or not. By using the original car insurance network, direct comparisons could be made between the elicited CPTs and the gold standard CPTs.

The online survey was conducted using the BNE software. It was extended from the software used in Chapter 5 to support the CPT part of SEBN described in Chapter 6.

The CPT elicitation evaluation took place over a period of approximately two weeks in 2013. All participants were recruited in this time window, no recruiting was done prior to the survey starting.

7.1.2. Recruiting Participants for Survey

The recruitment of participants for this evaluation was similar to that of the previous evaluation for structure elicitation (Chapter 5). The choice of participants was discussed in Section 5.1.4 (p106) and how many questions to allocate each participant was discussed in Section 6.3.3, (p170).

This evaluation used a different set of participants from the previous. As with the structure evaluation, this evaluation left the number of participants (n_e) as variable in the hope that as many people as possible could sign up. The number of questions asked of each participant (n_q) was fixed.

Section 6.3.1 (p168) outlined how the number of parents for a given variable dictates the types of questions which need to be asked to elicit the CPT for that variable. The BNE software was configured to allocate groups of questions to each participant, such that answering each of the questions in a group would result in the CPT for a given variable. Participants were allocated multiple groups until n_q exceeded 60. Based on an earlier ex ante evaluation while developing the BNE software, it was estimated this would require approximately 15 - 30 minutes to complete.

Figure 7.2 shows that of the 64 participants who created an account for the BNE software, 56 consented to participate in the survey. Of these, 50 completed *at least one* group of variables resulting in a CPT, while 44 completed *all* allocated questions including a short demographic survey at the end. Participants who completed the entire survey were given the choice of entering the draw for a single $100 cash prize. All 56 consenting participants opted to go into the draw for the prize.

The spread of responses across variables is shown in Figure 7.3. This shows the number of participants who responded sufficiently to produce a full CPT for each variable. The lowest number of CPTs elicited for a particular variable was the *SocioEcon* variable which had CPTs elicited from three different participants. The

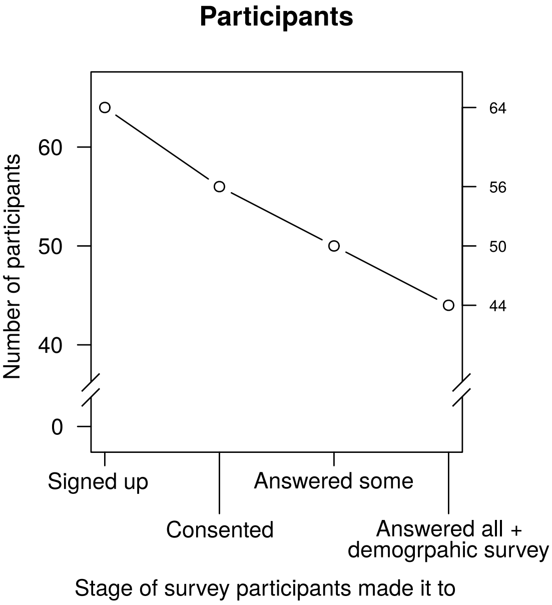


Figure 7.2.: Visualising the participation dropout as the survey progressed.

highest number of elicited CPTs was seven for the *Accident*, *DrivinkSkill*, and *MakeModel* variables, each with 7 CPTs. The spread is similar to that in the structure elicitation survey (Figure 5.3, p110).

Participant Demographics As was discussed in Section 5.1.3 (p105), the car insurance network (Binder et al., 1997) was chosen as the gold standard in part because the concepts depicted in it should be familiar to most people who drive

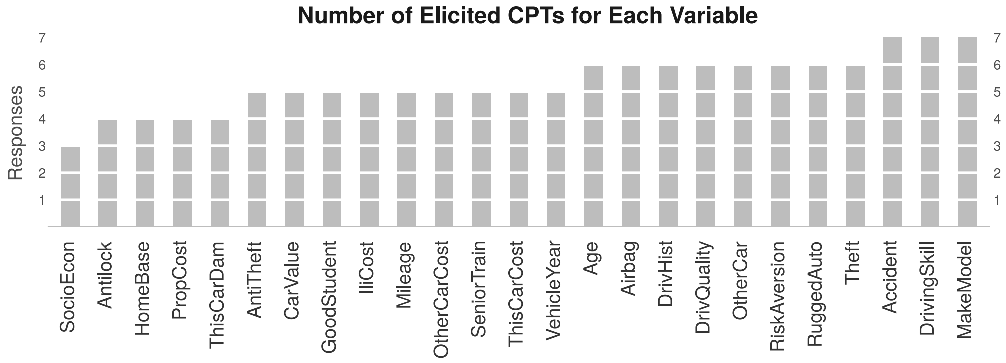


Figure 7.3.: Number of participants who completed questions about each CPT.

cars. To get an insight into the participants used in this evaluation, and whether they could reasonably be expected to be familiar with driving a car, the BNE software was configured to ask participants about their driving habits at the conclusion of the elicitation survey. The results (shown in Figure 7.4) indicate that a large proportion of respondents were familiar with driving cars, many of them having held a drivers license for over 5 years. Also, the vast majority will likely have experience with similar driving conditions to each other due to living in Victoria, Australia.

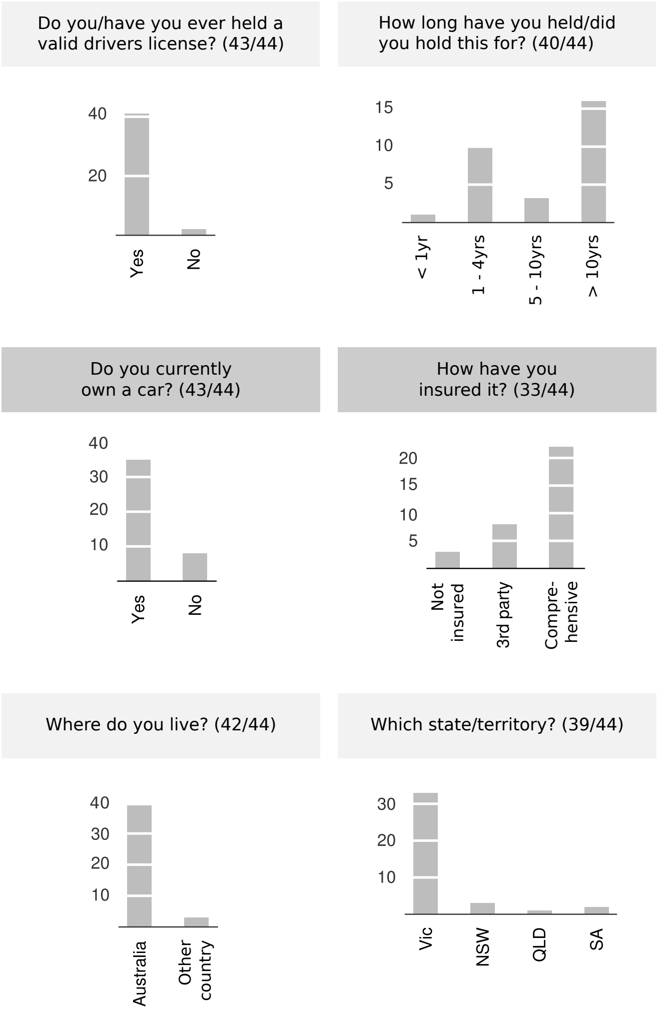


Figure 7.4.: Results of the demographic survey. Numbers in parenthesis are the proportion of participants who answered the question.

Label	Description
Gold	"Gold Standard" Existing network from literature (Binder et al., 1997).
Mean	"Mean" Average all CPTs for a node, then normalize.
DS	"Dawid & Skene" Collate using EM algorithm (Dawid and Skene, 1979; Project Troia, 2013) then normalize.
Survey	$\{Mean, DS\}$
$Learnt_{Bayes}$	Learnt from *the bnlearn* software, via the `bn.fit` function with `bayes` and `mle` algorithms respectively. Uses data sampled from the *Gold* network.
$Learnt_{MLE}$	
Learnt	$\{Learnt_{Bayes}, Learnt_{MLE}\}$
Eval	$Survey \cup Learnt$

Table 7.1.: The different algorithms used to produce CPTs for evaluation.

7.2. Comparing CPTs of Individual Variables

This chapter will produce a number of different CPTs in order to compare those to the gold standard car insurance network (Binder et al., 1997), and others learnt using the *bnlearn* software (Scutari, 2010).

7.2.1. Collating Survey Responses into CPTs

Before analysing results from the online survey they must first be collated into a single CPT for each node in the gold standard BN structure. This section describes how the responses from multiple participants were collated to form the requisite CPTs.

Overview of the Different CPTs Being Evaluated Table 7.1 shows the different ways in which CPTs were elicited from the survey. It also shows others which were learnt from existing algorithms solely for the purpose of evaluating this research.

Mean and *DS* algorithms both start with full CPTs output from the BNE survey software (as opposed to raw, unprocessed survey responses), after applying the weighted sum algorithm where relevant. Thus, for each variable in the BN there is multiple CPTs for which to combine. The *Mean* network collates CPTs by averaging the value for each conditional probability across participants, then normalizing the CPT. The *DS* CPTs are collated using the EM algorithm from the *Troia* software package (Project Troia, 2013) and then normalized.

The *Learnt* CPTs were created using existing algorithms provided by the *bnlearn* software (Scutari, 2010). A training data set was sampled from the *Gold* network ($n = 1000$) and then input to the `bn.fit` function using `bayes` and `mle` algorithms to produce the CPTs. Note that these algorithms were unable to deal with variable states that were highly unlikely or impossible. The states in the *Gold* network with a probability of 0.0 such as $\Pr(OtherCarCost = Million|Accident = None)$ were never represented in the training data set and those with probabilities approaching 0.0 such as $\Pr(Theft = True|HomeBase = Secure)$ were very unlikely to be represented in the data. While it is possible to increase the number of sampled records in order to ensure that unlikely states are observed, this is undesirable. Firstly, improvements from the increase in training data diminish as the size of the data set increases. Secondly, there will *always* be *some* states which are never observed due to their probability being 0.0 and so the problem will still persist no matter how many records are sampled for the training data. The way in which `bayes` and `mle` algorithms from the *bnlearn* software handle unobserved states differs. The `bayes` algorithm sets the CPT for the unobserved states to the uniform distribution. The `mle` algorithm sets them to `NaN` values. This chapter will highlight when this impacts on the analysis.

Selecting CPTs to Evaluate A subset of all variables from the *Gold* BN have their CPTs evaluated in this chapter. The variables which are evaluated are shown in Table 7.2. The variables which were excluded are: *SocioEcon, OtherCar, GoodStudent, RiskAversion, AntiTheft, HomeBase, VehicleYear, MakeModel, RuggedAuto, Antilock, Airbag, Theft, ThisCarCost, MedCost,* and *Cushioning.* Appendix D (p257) provides a more comprehensive discussion on the choice of

Variable	Number of Parents
Age	0
Mileage	0
ILiCost	1
DrivingSkill	2
PropCost	2
SeniorTrain	2
DrivQuality	2
DrivHist	2
ThisCarDam	2
OtherCarCost	2
Accident	3

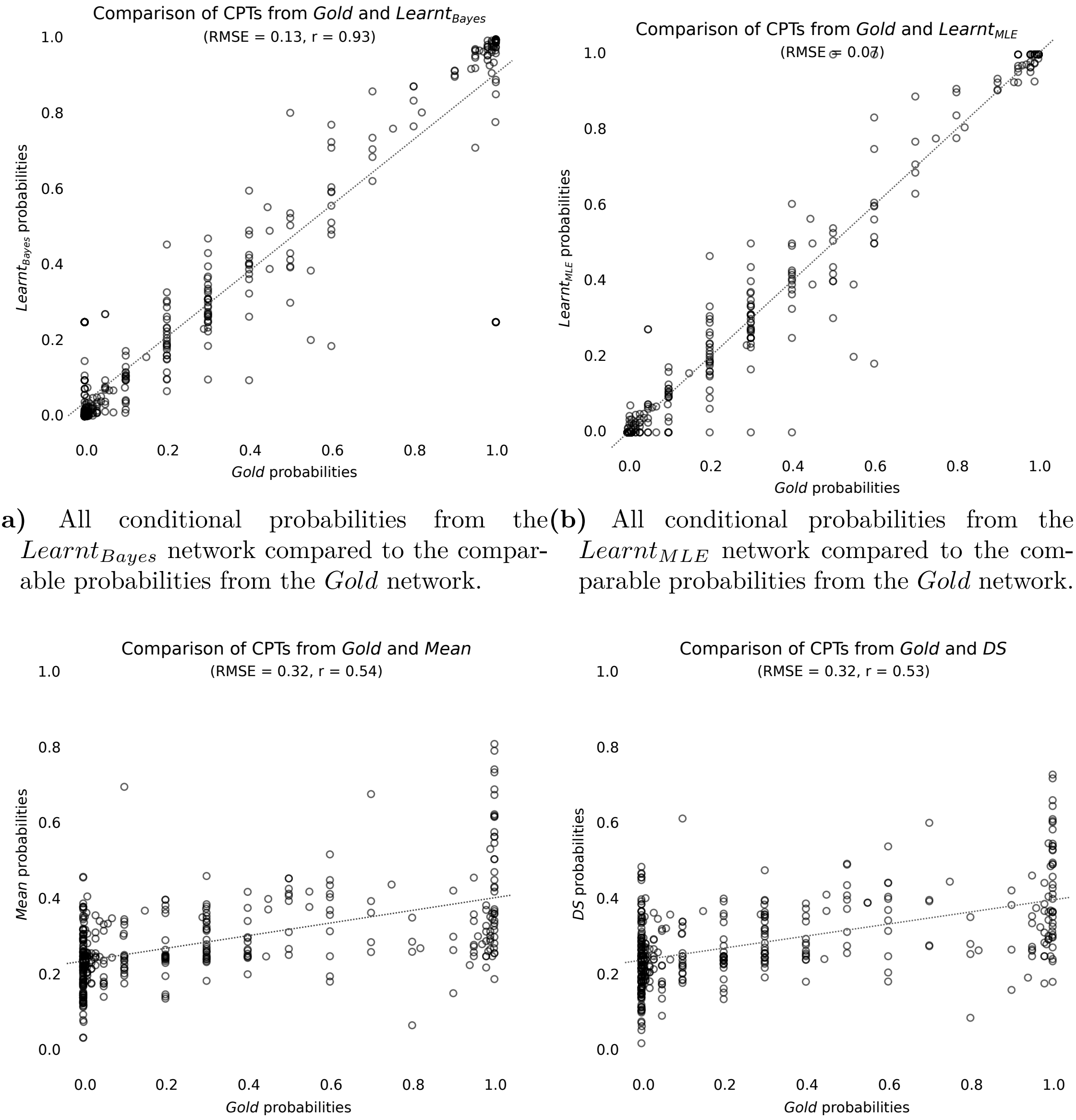
Table 7.2.: The variables which are evaluated in this chapter.

which CPTs were evaluated, and which were left out (and why).

7.2.2. Summary of Probabilities

Figure 7.5 shows conditional probabilities from the *Gold* network and compares them to the same conditional probabilities from the *Survey* and $Learnt_{Bayes}$ networks. The probabilities from the $Learnt_{Bayes}$ CPTs show a very strong positive correlation with those from the *Gold* network (r = 0.93, Figure 7.5a). The *Survey* networks contain much more uniform probability distributions compared to those in the *Gold* network. However, they are both still positively correlated (r = 0.54, Figure 7.5c and Figure 7.5d). Note the outliers at $Learnt_{Bayes} = 0.25$ (both when $Gold = 0.0$ and $Gold = 1.0$) which are due to the training data used to learn $Learnt_{Bayes}$ not containing any observed values of these states.

The CPTs from $Mean$ and *DS* are quite different to those from *Gold* despite being positively correlated. This can be seen by analysing the RMSE of the *Eval* CPTs compared to those from the *Gold* network (Figure 7.6). There is a substantially larger error in the *Survey* and *DS* CPTs than those of the *Learnt* networks due to the uniformness of the probabilities in the *Survey* CPTs observed in Figure 7.5c.



(a) All conditional probabilities from the $Learnt_{Bayes}$ network compared to the comparable probabilities from the *Gold* network.

(b) All conditional probabilities from the $Learnt_{MLE}$ network compared to the comparable probabilities from the *Gold* network.

(c) All conditional probabilities from the $Mean$ network compared to the comparable probabilities from the *Gold* network.

(d) All conditional probabilities from the DS network compared to the comparable probabilities from the *Gold* network.

Figure 7.5.: Comparison of the probabilities of the *Gold* network compared to the equivalent probabilities for the $Mean$, DS, and $Learnt_{Bayes}$ networks respectively.

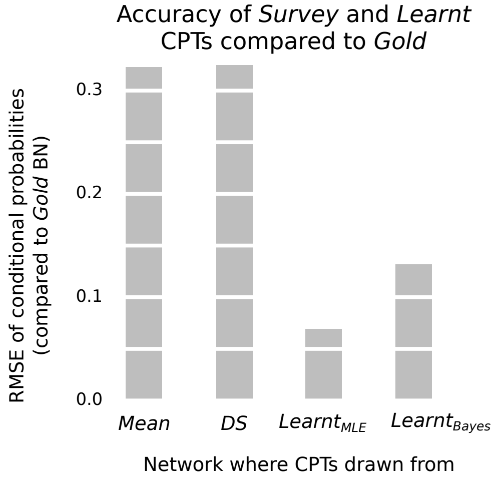


Figure 7.6.: Root Mean Squared Error of the conditional probabilities in $Mean$, DS, $Learnt_{MLE}$, and $Learnt_{Bayes}$ when measured against the probabilities in *Gold*.

7.3. Verifying Coherency Among Responses

While building a system to elicit CPTs from experts, Hope et al. (2002) discussed the concept of coherency. The sum of each elicited probability conditioned on the same sample space must sum to one in order to adhere to one of the basic axioms of probability theory. The greater the distance from one, the less coherent each individual probability is likely to be. The future work section from Hope et al. (2002) discussed how it would be useful to explore other mechanisms to normalize values while maintaining coherency. This research agrees that it would be helpful, and thus this section provides an analysis of how coherent responses tended to be in order to provide data for future researchers to work with.

Each point in Figure 7.7 shows the sum of an elicited probability values for the different states of a variable, elicited by individual participants. For variables in the BN with only a single parent, each point refers to, e.g. $\sum_{x \in states(ILiCost)} \Pr(ILiCost = x|Accident = True)$. For variable with no parents, each point refers to the sum of all probabilities for each state of that variable, e.g. $\sum_{x \in states(Age)} \Pr(Age = x)$. They should sum to 1 to be completely coherent.

The end result shown in Figure 7.7 is a large bias towards overstating the probab-

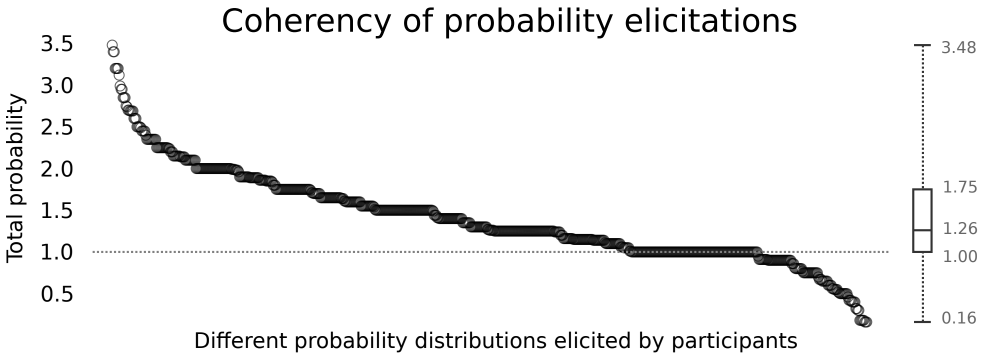


Figure 7.7.: Each point represents the sum of an individual participants elicited probability values for different states of a variable, conditioned on specific states of its parents.

ility of events. Almost all probabilities summed to > 1, with only a small number summing to 1 and an equally small amount understating the probability of an event ($\mu = 1.40, \sigma^2 = 0.55$). This shows that participants were not very coherent when responding to survey questions, or that the survey questions were crafted in such a way as to allow incoherent answers.

7.4. Time Spent by Participants Doing Survey

The analysis of time spent by participants during the survey was done by looking at the web server access logs, grouping users based on their IP address. The way in which the time of each session was calculated is discussed in Section 5.4 (p138). Although it is possible that multiple participants accessed the survey through the same IP address due to being behind a NAT router, this was not the case in this study. This is known because there was 44 different users who were deemed by the application code to have completed the survey. In corroboration with this, the web server logs show 45 unique IP addresses visiting the URL that corresponds to completing the survey[1]. This page could only be visited by logged in users who had completed the survey. Thus, it is concluded that for all participants who

[1] The discrepancy between 45 IP addresses and 44 users finishing the survey is likely due to an error occurring on one of these page visits, preventing the application from logging the users completion of the survey.

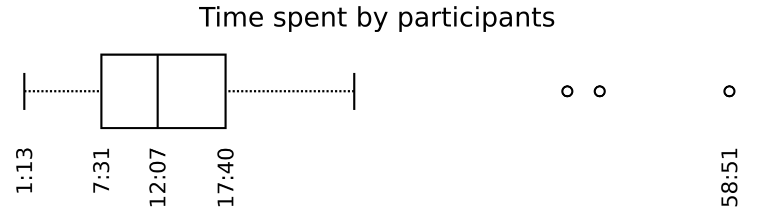


Figure 7.8.: Time spent by unique IP addresses participating in the online survey (though not necessarily finishing). Note that this is lower than the actual time spent by participants, due to the reasons discussed in Section 7.4.

completed the survey, they did not share an IP address.

In total, the web server logs had 69 unique IP addresses, 57 of which logged 10 or more page views in the application. Of these 57 unique IPs, 45 logged a page view for the URL indicating they completed the survey. This means that there is about 12 IP addresses which either belong to existing participants returning for another session, or participants who started but didn't complete the survey. Thus, the times shown in Figure 7.8 will be shorter than they should be, because some of those 12 sessions belong to the 45 other people. To cater for this, the median time of 19 minutes 13 seconds is taken from the sum time spent by all 57 IP addresses, but divided by 45 (the participants who actually completed). This may be slightly higher than the real median, because the 12 unaccounted IP addresses may well belong to participants who started but didn't complete the survey.

CPT elicitation is frequently referred to as the most time consuming aspect of BN elicitation (Druzdzel and van der Gaag, 2000). The results here show that the amount of time required of each participant in order to parameterise this particular BN via SEBN is minimal. Even after taking into account the maximum time of almost an hour, the inflated median of 19 minutes is still a small amount of time compared to traditional KEBN CPT elicitation sessions using face to face interviews.

7.5. Verifying Accuracy of Das Estimations

The weighted sum algorithm is based on the assumption that:

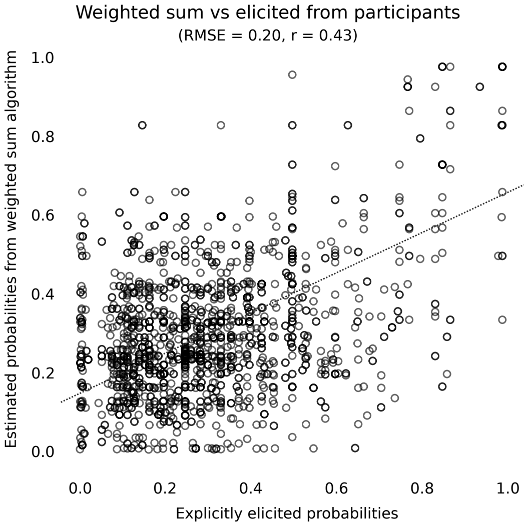


Figure 7.9.: Correlation between explicitly elicited conditional probabilities and their respective estimated value from the weighted sum algorithm (Das, 2004).

$$\Pr(Y = y|X_1 = x_1, ..., X_n = x_n) \approx \sum_{i=1}^{n} (w_i \times \Pr(Y = y|CPC(X_i = x_i)))$$

In the original paper (Das, 2004) the assumption behind the algorithm is justified using heuristics, information theory, and other arguments based on intuition. This section empirically evaluates this assumption with data from the evaluation survey.

For each variable with > 1 parent, the BNE software elicits responses $r_{prob} \cup r_{CPC} \cup r_{AHP}$. Each r_{prob} response entails an explicit elicitation of a conditional probability. This can then be compared to the estimation of the probability by providing $r_{prob} \cup r_{CPC} \cup r_{AHP}$ as input to the weighted sum algorithm. The results are shown in Figure 7.9. The analysis shows that there is a weak positive correlation between the elicited probabilities and those estimated using the weighed sum algorithm. Given the weak correlation coefficient of 0.43 and the RMSE of 0.20, further investigation is warranted to decide whether the weighted sum algorithm does in fact provide an appropriate way to estimate CPT values from sparse elicitations.

7.6. Chapter Summary

This chapter documented an online survey which was conducted to elicit the CPTs of a BN using SEBN described in Chapter 6 (p155). It collated responses from multiple participants to see how close the elicited CPTs were to that of a gold standard BN. The collation was done using both the mean, and the Dawid & Skene algorithm, to evaluate how each algorithm performed.

A successful evaluation would result in the CPTs collated from the online survey being similar to those from the gold standard (as measured by RMSE). Additionally, it should take less time of each participant compared to if the probabilities were elicited in one or more face to face interviews ala traditional KEBN.

The results from Section 7.2.2 shows that the CPTs elicited from the online survey do not compare favourably to the gold standard. They are also outperformed substantially by CPTs from the *bnlearn* software. Despite this, it is encouraging that the CPTs are positively correlated with those from the gold standard. The main difference from the gold network is that the CPTs elicited from the online survey tend to be more uniform.

In the process of conducting this evaluation, Section 7.5 took the opportunity to empirically evaluate the assumptions behind the weighted sum algorithm. It found that there was a very weak positive correlation between the explicitly elicited probabilities, and what the weighted sum assumptions predicted would be a suitable probability. This shows that further research is required in order to identify the situations in which the weighted sum assumptions are relevant, and when the algorithm can be used in order to reduce the magnitude of the CPT elicitation task.

The median time spent responding to the survey was approximately 20 minutes with a maximum time of about 60 minutes. Taken together with the previous evaluation for structure elicitation which showed between 30 and 60 minutes, the total time expected of each participant to participate in an online survey such as that conducted for these evaluations is approximately 1 to 2 hours. Eliciting the structure and the probabilities of a BN are two of the three required tasks, the other of which is eliciting variables to include in the model. Thus, although this does not include all aspects of eliciting an entire BN, 1 to 2 hours is a significant contribution towards that goal.

8. Conclusions and Future Work

This thesis proposed and evaluated a method for eliciting BNs from experts using online surveys in place of interviews. The conclusions of the research are discussed in this chapter.

8.1. Thesis Summary

This thesis was motivated by a comparative lack of research in the area of knowledge elicitation, compared to machine learning methods for constructing BNs (Section 1.1, p3). The goal was to utilise online surveys to reach a larger and more diverse pool of experts to contribute their knowledge to the elicitation process. Online surveys were chosen because they could be conducted at the experts leisure, and the responses can be collated together with little effort from the researchers. This method of BN elicitation was termed SEBN, as opposed to traditional KEBN (Korb and Nicholson, 2011, p297). It was evaluated using two studies, whereby a BN was elicited using an implementation of the online survey system. The resulting BNs were compared to a gold standard network, to see how similar they were and how long was required of each participant to conduct the online surveys.

Literature Review The literature review in Chapter 2 provided background on why BNs are desirable models, and the ways they are currently constructed. This was followed by an in depth look at both surveys and traditional KA. In particular, it discussed the various biases that tend to be exhibited when eliciting knowledge through interviews, or when collecting data via surveys. In the interests of ensuring the minimal amount of effort was required to combine survey responses from multiple experts, the literature review also looked into various ways in which knowledge from multiple people can be combined. This included the process of combining knowledge from multiple experts participating in focus groups, and the field of crowd sourcing, which combines multiple responses from lay people. This is important for combining survey responses about BN elicitation into one authoritative BN.

Methodology The Design Science Research (DSR) method was used to frame the research conducted for this project, and was detailed in Chapter 3. There were two main artifacts to arise from this research. The first is a method for eliciting BNs using online surveys in place of interviews (SEBN). The second is an open source implementation of this method (BNE). Before designing and implementing the system, a set of propositions was outlined which detailed the expectations of such a system. Among other things, these propositions stated that the method should:

- Reduce the burden on experts and researchers alike.
- Output a useful BN at the end of the process.

Building BN Structure Through Survey Based Elicitation The method for eliciting the structure of BNs using online surveys was presented in Chapter 4. The method revolves around converting all of the possible relationships between different variables into causal questions of the form “Does X influence Y?”. This results in n^2 questions (where n is the number of variables to include in the BN), which is too many to satisfy the stated goals of reducing the burden on experts. To address this, variables are first categorised into classes in the manner described by Kjærulff and Madsen (2013). This ensures that, for example, problems are able to cause symptoms but symptoms are never the causes of problems. The second way in which the burden on experts was reduced was by only allocating a subset of all possible questions to each expert. This drastically reduces the effort required of each expert, however it means that individuals are not exposed to the entire problem space. This may result in anomalous relationships appearing in the BN structure once collated, which is undesirable. To counteract this, Chapter 4 discusses how to deal with anticipated problems that may arise due to this phenomena. The most problematic of these is the presence of cycles, which prevent a valid BN structure from being formed. Algorithms for removing cycles are discussed in Chapter 4, and also elaborated on in the Future Work section of this chapter (Section 8.7.8, p218).

Calculating BN Probabilities Through Survey Based Elicitation The elicitation of CPTs via online surveys was documented in Chapter 6 (p155). CPT elicitation is particular sensitive to combinatorial explosion and usually requires a large number of probability estimates to be elicited (van der Gaag et al., 1999). This chapter addressed the problem in three phases, nodes with zero parents, one parent, and multiple parents. For nodes with zero or one parent, each of the required probabilities is elicited explicitly using a question of the form “What is the likelihood of the following scenario?” or “What is the likelihood of the following scenario, if we know that ...?” respectively. The individual probabilities are elicited using “The Fragment of Text and Probability Scale for the Assessment of Conditional Probability” adapted from van der Gaag et al. (1999). Nodes with multiple parents are treated differently, as the number of required probabilities required to populate a CPT grows exponentially with the number of parents a node has. Thus, Chapter 6 presents a method for eliciting only partial CPTs using online surveys, and then interpolating the remaining values as per Das (2004) and Saaty (1977). As with Chapter 4, the CPT elicitation only allocates a subset of all questions to any given expert participating in a survey to reduce the time required of them.

Evaluating Structure Elicitation and Evaluating Probability Elicitation After Chapter 4 and Chapter 6 proposed SEBN as a technique for using online surveys to elicit BNs, Chapter 5 and Chapter 7 discuss the experimental evaluations which took place. The goal of these chapters was to collect data in order to answer the propositions detailed in Chapter 3 to see if they came to fruition or not. The evaluations started with an existing BN from the published literature, referred to as the gold standard network. Then, two online surveys were constructed using the BNE software. The first was to elicit the structure of a BN and the second was to elicit CPTs.

The structure elicitation survey enlisted 43 participants, of which 23 completed the questions allocated to them. The evaluation then collated these responses into BN structures using both the majority vote, and the EM algorithm. These two algorithms were compared to see which was better able to collate responses in such a way that they corresponded more closely to the gold standard network. For reference, they were also compared to three existing algorithms for learning BNs from data, and two nonsense BNs to provide a lower bound on what is considered acceptable. The CPT elicitation survey enlisted 64 participants, of which 44 answered all of the questions allocated to them. Similar to the evaluation of the elicited BN structures, the elicited conditional probabilities were collated using both the mean, and the EM algorithm.

Unfortunately, the results showed that the BN structure and CPTs elicited using the survey technique were not comparable to either the gold standard, or the existing algorithms for learning BNs from data. Section 8.7 provides some directions for future research which should improve the survey method, such that it is able to compete with other elicitation techniques and provide an alternative that puts less burden on researchers and experts alike.

The positive out of the evaluation was that although the resulting BN did not compare favourably to the gold standard, SEBN was indeed able to facilitate the elicitation of a BN structure and its associated CPTs. It conducted the surveys in less time than would have been required for traditional KEBN. As such, future work may improve on SEBN with the confidence that if the elicitation of BN structures and CPTs becomes more robust, then it will be a viable alternative to traditional KEBN capable of reducing the burden on experts, and thus facilitating the elicitation of knowledge from a broader range of stakeholders.

8.2. Revisiting the Propositions Post Evaluation

This section will evaluate the specific propositions described in Section 3.3 (p60). The aim is to use the results of the evaluation studies described in Chapter 4 and Chapter 6 to evaluate if the relevant propositions held true.

8.2.1. Measuring Time Commitments (Proposition 1 & 2)

Proposition 1 and Proposition 2 state that SEBN will require less time of experts and researchers alike. This evaluation seeks to ascertain whether a BN elicitation project using SEBN can reasonably be expected to take less time than a traditional KEBN elicitation project.

Time of Experts

There were two main evaluation surveys undertaken for this project. Prior to these, an earlier structure elicitation evaluation based on an earlier iteration of SEBN was abandoned due to the amount of time required of experts who were contributing. Those experts found that there was too many comments required, and that they were not willing to spend the requisite amount of time committing to the project. In response to this, the method outlined in Chapter 4 and Chapter 6 emphasizes allocating only a subset of questions, resulting in less time required of participants.

During the evaluation survey in Chapter 5, the median time spent by participants answering questions for the BN structure elicitation survey was less than 30mins, and the maximum time spent was less than an hour. The CPT evaluation survey in Chapter 7 resulted in a median time spent of under 20mins, with a maximum of just under an hour.

In addition to reducing the time requirement of experts by only allocating a subset of questions, they did not need to travel in order to come to an interview. This could also be the case for traditional KEBN interviews conducted at the experts place of work, but it is not the case for other types of interviews.

	Preparation	Recruiting	Administration	Transcription	Analysis
Surveys	●●●	●●●	●○○	●○○	●○○
Interviews	●●○	●●○	●●●	●●●	●●●

Table 8.1.: Summary of researchers time expected to be spent on traditional KEBN and SEBN.

Time of Researchers

This section addresses how long was spent by the researchers while setting up, administering, and analysing results from SEBN elicitations. It seeks to identify and focus on points at which significant input is required by the researcher, and highlight those where less time is required. The summary of these findings are presented in Table 8.1.

Recruiting This needs to be done both for traditional and for survey based approaches. For this evaluation, participants were recruited by talking to colleagues in person, and also by contacting extended friends via an online social network (Facebook). For the CPT elicitation evaluation, advertisements were shown on Facebook for the duration of the survey encouraging people to sign up and participate[1]. In terms of time spent, it was comparable to recruiting experts for a study using traditional KEBN.

Variable Elicitation Variables were identified based on their presence in a gold standard BN. Thus, the process of variable elicitation was effectively skipped. This would not be the case with a typical SEBN project. Given that Chapter 4 did not define how variable elicitation should be accomplished for SEBN, it is likely that it would be conducted in the same way as a traditional approach. However, during interviews variable elicitation is done at the same time as the rest of the process whereas a survey approach would likely require an additional survey. This is because all variables need to be available before the structure elicitation survey can begin. Thus, the variable elicitation will likely take longer for SEBN compared to traditional KEBN. The potential

[1] The Facebook ads were targeted towards those living in Australia, older than 18, who speak English and have an interest in one of the following: Automobiles, Insurance, SurveyMonkey, NASCAR, Education, Driving, Vehicle insurance, Teacher, Survey data collection or Motor sport.

for adding support for variable elicitation to SEBN to further reduce the burden is discussed in Section 8.7.4 (p216).

Configuring Software For this evaluation, time was spent setting up the list of variables and their descriptions. This allowed the system to subsequently generate questions as per Section 4.2, p75 and Section 6.2, p156. This time can be seen as similar to the time used preparing semi-structured interviews or focus groups before actually conducting them. The future work (Section 8.7.11, p220) discusses how this could be made easier with more refined software, or via a hosted online service similar to SurveyMonkey, LimeSurvey, or other more traditional providers of traditional surveys. This would reduce the amount of configuration required by the researcher. It also would remove the requirement to setup a web server, install the software, make it publicly accessible to the internet, etc. It would *not* remove the need to configure the software with the variables of interest, or customize things such as the consent form, etc.

Administering Survey/Conducting Interview The total time spent administering the survey was negligible for the survey. Once recruiting is completed, then the administration time consists of the time spent compiling email reminders to participants who had not completed the survey. A comparative interview in traditional KEBN would be longer, and demand the constant attention of the researcher over the period of time the interview is being conducted.

Analysing Results There was close to zero time required to analyse the survey results. To move from administering the survey to collecting responses and producing a BN, it was a matter of pressing a button in the BNE software (and waiting a few minutes for it to complete). The comparable time spent in traditional KEBN would be much longer, although much of it would likely have happened during the interview rather than afterwards. Despite the different point in time at which the BN is constructed, the time required for survey elicitation is much shorter than traditional KEBN.

Was Proposition 1 & Proposition 2 Met?

> *Yes* - Although the reduction in time commitments is greater for experts than it is for the researcher.

8.2.2. Geographically Dispersed Participants (Proposition 3)

Proposition 3 states SEBN should be more suitable for geographically dispersed experts. The experimental evaluation conducted here made use of an online survey discussed in Section 5.1 (p104). It consisted primarily of people from Melbourne, Victoria, Australia. Having said this, there is nothing about the online survey or the administrative process for selecting participants which restricted the origin of the participants. Indeed, over 10% of the participants in the CPT elicitation survey identified as living interstate (Qld, NSW, and SA) and three overseas, and this did not impact their ability to participate.

One concern that may arise when using participants from diverse timezones is that they will find it difficult to commit at the same point in time. This was not an issue during the evaluation of SEBN, because data collected in a highly asynchronous manner. The two evaluation surveys were online for a period approximately one month and two weeks respectively, with participants free to contribute at their leisure.

If, for administrative or other reasons, the survey needed to be completed in a matter of hours or days, then geographically disperse participants are *not* suitable if the difference in time zones is substantial. Indeed, one of the reasons for proposing this research was so that experts did not need to congregate at the same time (and place) in order to participate. It then follows that requiring people to get together at approximately the same time is not one of the goals of the survey software, and thus perhaps traditional elicitation is more appropriate (though it would require some form of video/voice conferencing to deal with the geographically dispersed participants).

Was Proposition 3 met?

> *Yes* - as long as the survey does not need to be completed in less than a single day.

8.2.3. Distinguish Level of Expertise (Proposition 4)

Proposition 4 (p63) talks about the ability of SEBN to better distinguish between differing levels of expertise in participants. Section 5.5 (p140) investigated and concluded that the Ipeirotis et al. (2010) algorithm was not able to distinguish high performing experts from those who answered less in line with the gold standard. The Future Work chapter of this thesis discusses the potential for incorporating other, more recent crowd sourcing algorithms to collate survey responses (Section 8.7.6, p217). Some of these may be better able to estimate the accuracy of experts.

Was Proposition 4 Met?

> *No* - The survey method was not able to distinguish better or worse performing experts.

8.2.4. Traditional Method More Flexible (Proposition 5)

The evaluation was not able to empirically measure this due to the evaluation only conducting an online survey and not a traditional elicitation process (opting to use a gold standard instead). Nevertheless, it is worth noting that during evaluation, the survey questions were intentionally kept the same after beginning, in order to not interfere with the process. This in itself hints at the fact that SEBN is not as flexible as a structured interview type approach.

In order to collate responses from many different experts over a period of a month, the responses given by different experts needed to be to the same question. Otherwise, it would be difficult to know if different responses were due to individual experts having different opinions, or due to a change in the question being asked of each expert. This came at the expense of some minor confusion during the evaluation. When one participant explained that they didn't understand one of the questions, it was too late to change the wording of it as others had already answered that question in its current form. In a traditional KEBN setting, the question could've been rephrased. This would not only clarify for the expert who is having difficulties, but also potentially for experts in the future who may struggle with the same question.

Was Proposition 5 Met?

> *Yes* - The survey method does not encourage changing questions midway through an elicitation, whereas traditional KEBN does.

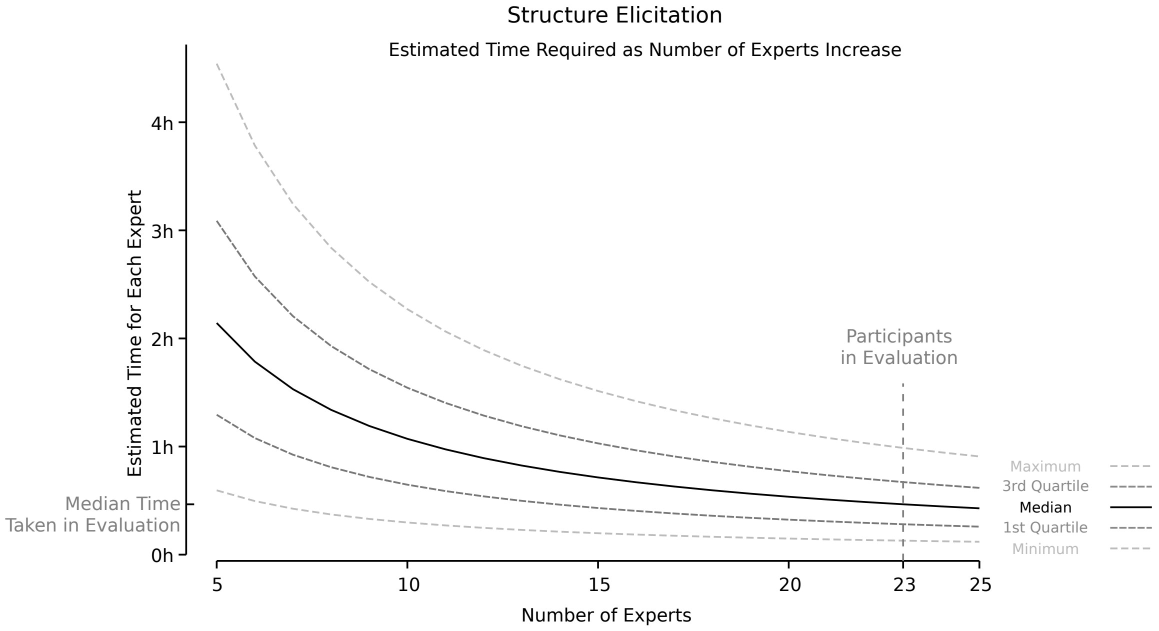
8.2.5. Traditional Method When Less Experts (Proposition 6)

There are a few ways to approach measuring if this is the case. The first is to see how long each expert takes to contribute to SEBN, and then model how much longer would be required if there were less experts participating. The second is to perform traditional KEBN with a single expert, and then add more experts and repeat, in order to model how much longer would be required if more experts are participating. This evaluation looked at the first approach.

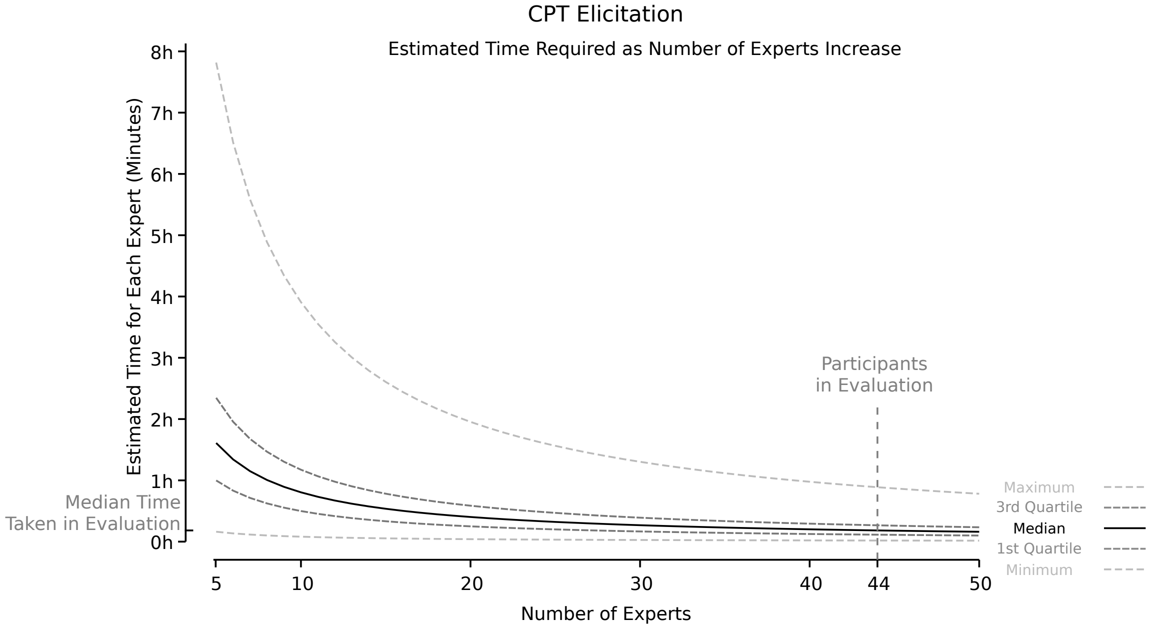
Measuring the time spent by experts was discussed earlier in Section 8.2.1. The number of questions allocated to each expert is covered in both Section 4.4 (p85) and Section 6.3 (p168).

Figure 8.1 shows the exponential increase in estimated time required of each expert as the total number of experts decreases. These values are modelled based on structure and CPT SEBN evaluations conducted during this thesis. For both structure and CPT elicitation, the estimated time required increases quickly as the number of experts drops below five. Above that, the median expected time is approximately 2h for each survey, totalling 4h for the elicitation of the structure and CPTs of a BN. For a BN with 25 nodes such as the one evaluated in this thesis, anything less than about five participants increases the time required well beyond 2hrs per expert.

In addition to measuring the time burden of each expert as in Figure 8.1, the evaluation also looked at artificially removing experts from the response pool in (Section 5.3.4, p136). The resulting networks collated from a smaller number of survey responses were indeed of much less quality than those collated from more responses, although the ML algorithm helped mitigate this somewhat.



(a) Estimated time required of each expert for *structure* elicitation survey.



(b) Estimated time required of each expert for *CPT* elicitation survey.

Figure 8.1.: Exponential increase in estimated time required of each expert, when less experts are enlisted to elicit the structure of a BN.

Was Proposition 6 Met?

> *Yes* - The survey method has an exponential increase in time required of each expert as the total number of available experts decrease. Also, the quality of the elicited networks degrade as less expert responses are collated.

8.2.6. Not Identical, but Similar Structure and Score (Proposition 7, 8 & 9)

The evaluations in Chapter 5 and Chapter 7 measured quantitatively how well the BN structures and CPTs elicited via SEBN compared to the original *Gold* standard.

As shown quite comprehensively in Section 5.3 (p122), the structure of the elicited BN structures were not at all similar to the *Gold* network. It was poor both in terms of the SHD as well as the ROC and F1 metric, and the BIC score of the *Survey* networks was poor, being outscored by all three *Learnt* network structures. In addition, Section 7.2.2 (p182) discussed how the CPTs elicited were inferior to those from the *Learnt* networks.

It is hoped that by addressing the issues raised in the Future Work section of this thesis (Section 8.7), SEBN can be refined until it can be used to elicit BNs that are better able to model the desired probability distribution. In addition, the method at its current state of research and development may still be found useful as a tool for constructing prototype BNs for low cost, as part of larger elicitation projects.

Was Proposition 7, 8 & 9 Met?

> *No* - The evaluation did not establish sufficient evidence that online survey elicitation can produce network structures that are similar, or result in a similar probability distribution, as the *Gold* standard network.

8.3. Addressing the Research Questions and Goals

The introduction chapter outlined the following research question in Section 1.2 (p5):

> *How can the process of eliciting knowledge for construction of Bayesian Networks be improved by making use of online surveys instead of face-to-face interviews?*

This was composed of the following two sub-questions:

1. *As more experts are consulted, how can the total time and effort involved in KA for BNs be reduced?*
2. *As more expert opinions are gathered, how can they be collated into a single BN model without significantly increasing the workload to resolve differences?*

These questions revolved around the ability of SEBN to reduce burden and increase diversity of the stakeholders. However, even if they were all thoroughly successful, they are not of much use unless the method is able to produce usable BNs. As such, this section will also address the *failure* of SEBN to produce useful BNs during evaluation.

This section will take a bottom-up approach and first address the two sub-questions, culminating in a more general discussion about the primary research question. It will then discuss how the propositions (outlined in Section 3.3, p60, and revisited above were unable to result in useful BNs).

8.3.1. Addressing the Sub-Questions

The first question was:

> 1. *As more experts are consulted, how can the total time and effort involved in KA for BNs be reduced?*

This was answered in great detail in Chapter 4 and Chapter 6. Specifically, Section 4.3 (p80) discussed how variables can be classified using the method of Kjærulff and Madsen (2013) to exclude questions from the entire survey. This ensures that some clearly non-causal relationships are never presented to the experts. The number of questions required for CPT elicitation was addressed in Section 6.2.3 (p162) by using the weighted sum Das (2004) and AHP Saaty (1977) algorithms. Section 4.4 (p85) and Section 6.3 (p168) discuss allocating experts a subset of all questions, allowing them to contribute a small part of the bigger solution, rather than an entire BN by themselves. This is a different approach than previous elicitation techniques (e.g. Flores et al., 2011; Xiao-xuan et al., 2007), and several issues had to be addressed to ensure that the final BN model is not negatively impacted by each expert only being exposed to a subset of the entire solution space. Section 4.6 talked in depth about how to address issues which may arise due to each expert working on a small portion of the model, specifically cycles in the resulting BN structure, and also potentially indirect relationships. The result of all this is that the total time and effort of the knowledge engineer and experts alike was kept reasonable during SEBN.

> 2. *As more expert opinions are gathered, how can they be collated into a single BN model without significantly increasing the workload to resolve differences?*

Given the motivation of including a higher number of experts in the elicitation process, it is important to keep their workload down when trying to incorporate their many opinions into a final model. Traditional KEBN typically places the burden on the interviewer or researchers to figure out how to combine the varying opinions of participants. Section 4.5 (p90) and Section 6.4 (p171) showed how algorithms from the field of crowd sourcing can be incorporated into the survey process. This allows for automating the process of combining multiple opinions, while taking into account the varying capacity for each participant to answer.

The ML algorithm, inspired by earlier work done by Dawid and Skene (1979), was evaluated thoroughly in Chapter 5, comparing it to the more naive and more regularly employed majority vote algorithm. Unfortunately neither performed particularly well, which indicates that SEBN itself is in need of refinement. Section 8.7.6 discusses the prospect of incorporating more complex algorithms than ML to collate answers in an even better manner. This could build upon a recent proliferation of research into the field of crowd sourcing spurring on more research into these algorithms.

8.3.2. Addressing the Main Research Question

The main research question asks about improving the knowledge elicitation process for BNs by utilising surveys. As can be seen when addressing the sub questions above, the usage of SEBN in preference to traditional KEBN allowed a large number of experts to participate. This occurred without a corresponding increase in the amount of commitment required from experts when answering, or the researchers when constructing the resulting BN.

Despite the best efforts of this project, the evaluation in Chapter 5 and Chapter 7 has shown that SEBN at its current state of evolution was unable to produce BNs which compared favourably to the gold standard network chosen for evaluation. On reflection, this is likely partially due to the choice of participants who were not recruited from the car insurance industry, but rather were lay people with experience driving. The reasons behind this decision are discussed in great detail in Section 5.1.4 (p106), but it seems to have not been a suitable choice given the outcome of the evaluation. Also, as with any software implementation, it would be beneficial to have more opportunities to refine the software and perform more user testing. This thesis did perform two preliminary ex-ante surveys before the ex-post evaluation surveys discussed in Chapter 5 and Chapter 7. Each of these provided information that was subsequently incorporated into SEBN and also the BNE software. The experience of running the experimental evaluation also resulted in yet-more lessons, which should be incorporated into the software in the future to ensure experts are able to contribute successfully (see Section 8.7.11, p220).

However, the method did successfully address many of the propositions from Section 3.3 (p60), showing that it does indeed reduce the burden on experts, and facilitate a greater number of people to participate in the elicitation process. It is hoped that with further research into using online surveys in place of interviews, the quality of BNs that are output from SEBN will increase. Outlined in Section 8.7 (p214) are suggestions for future research and development which may be able to push SEBN in the direction of a suitable alternative for traditional KEBN.

It is strongly believed that if the method can be enhanced in the future, to the point where all of the propositions outlined in this thesis are met, then it will be a big step forward in BN construction, allowing a large number of people, organisations, businesses, to be able to make use of BNs when they previously would not have been able to.

8.4. Reflections on Methodology

While conducting the research for this thesis, it became apparent that there was a gap between the process of designing SEBN, implementing it in software, and then using it in an evaluation study. Lukyanenko et al. (2014) argue that IS researchers producing designs for artifacts should make explicit any concerns pertaining to moving from design to implementation, which they discuss using the framework of Instantiation Validity (IV). This is similar to researchers considering construct, content, predictive, reliability, inter-rater, and other forms of validity. The concerns raised by Lukyanenko et al. (2014) are that very few IS research articles discuss how to assert that a particular software implementation of a design or theory is a truthful representation of that design or theory. In other words, how do researchers ensure that the software they implement for a particular design has all of the benefits that the design claims to possess?

The conclusion from the Lukyanenko et al. (2014) paper is that:

> "Unless sound criteria for evaluating instantiation validity of IS design research is applied, doubts remain whether results are due to extraneous factors or attributable to idiosyncratic software development".

To address this, Chandra et al. (2015) suggested that researchers ensure any designs resulting from DSR projects have both *materiality* and *action* oriented design principles. Materiality oriented design principles focus heavily on how one would go about implementing a particular design. The action oriented principles focus on how a design may impact a humans life when used. IS research should present a healthy balance between both materiality and action oriented design principles.

Arazy et al. (2010) and Lukyanenko and Parsons (2013) also discuss the level of implementation detail to be bundled with any theory arising from a DSR project. In the absence of detailed implementation specifications, DSR researchers are unable to make any strong claims about IV. For this reason, SEBN in this thesis was presented in a prescriptive manner (Chapter 4 and 6).

8.4.1. Existing Threats to Instantiation Validity

Lukyanenko et al. (2015) discuss these five threats to IV and strategies to mitigate them. Each threat relates to a problem that arises when implementing an abstract theoretical concept as a tangible piece of software.

Threats Solved With More Resources

This thesis argues argues that three of the five specific threats identified by Lukyanenko et al. (2015) relate closely to the amount of resources that can be committed to implementing a particular design.

Artifact Cost This is related to the expensive nature of building non-trivial software. However, given enough time and money, it is possible to overcome this threat.

Artifact Instantiation Space The way in which many IS artifacts are specified is intentionally abstract, in order to highlight the theoretical implications over any implementation specific details. However, this means that the number of different ways in which a theory can be implemented is large, often intractably so. Spending more time eliciting requirements for a piece of software helps identify the important parts of the instantiation space to focus on.

Artifact Complexity The inherent complexity of some software systems make them difficult to implement, and thus risk introducing undesirable effects into any study making use of that artifact. However, more resources allow for better unit, functional, regression, and other forms of testing. This provides confidence that even if the artifact is complex, it works as expected.

Threats Not Solved With More Resources

Two threats from Lukyanenko et al. (2015) are less correlated with the resources committed to implementing a design.

Artifact Medium and Distance IS theories are specified in natural language, diagrams, and other explanatory memorandum. However, theories are implemented as software which is written in code and expressed in user interfaces. This disconnect has the potential to lessen the validity of conclusions drawn when evaluating implementations.

Technological Progress The norm in UIs and UX change over time. Although keeping pace with this progress is to some extent a matter of resources, entire changes in UI paradigms may require validity to be established again each time a new artifact is implemented.

8.4.2. Additional Threats to Instantiation Validity

This thesis offered several *materiality* oriented designs in Chapter 4 and 6, proving specific guidance to those wishing to implement SEBN. Despite this, there were other threats to IV which were not accounted for. This section presents two new threats to IV which arise when drawing conclusions from experiments which make use of software artifacts.

Many IS design theories are implemented in software for the express purpose of testing the IS theory in question. This usually consists of running an experiment, whereby participants are asked to perform some task with the newly implemented software. This is then compared to their ability to perform the task using their traditional approach. Results are examined to see if the software improves the ability of the participants to perform the task in question. When such an experiment is conducted, it is incumbent on the researcher to eliminate as many threats to validity as possible to ensure any results and therefore conclusions are not due to extraneous variables. It is desirable if any noted improvement in task performance can be attributed to the usage of the new artifact.

Two new threats to IV were identified while using BNE to evaluate SEBN. These were *Generally Technological Literacy* and *Familiarity with Specific Technologies* and they have the potential to negatively impact the conclusions of any experiment involving the implementation of a design theory. Both are closely tied to the field of Human Computer Interaction (HCI) when participants use the software in an experimental setting. Neither threat is completely averted by adding more resources to the implementation of SEBN. However, both can be addressed by ensuring adequate practice and training using BNE to participate in surveys.

General Technological Literacy

Most people will have varying levels of technological literacy. Some may be able to deal quite comfortably with most devices while others will struggle with each new device they encounter (e.g. phones, tablets, laptops, TVs, etc). Some may be able to adapt to different software for achieving the same task, while others are more comfortable sticking with the software they know (e.g. specific word processors, photo editing software, etc).

When implementing a software artifact and then conducting evaluations, it is important to consider the varying level of technological literacy for the target audience. This can be difficult, given that they will undoubtedly include users of varying technical literacy. Despite the best efforts of designers and developers, the most beautiful and usable software may not appear beautiful and usable to somebody who spends very little of their life using any sort of computer.

If a person who has never used a desktop computer before is asked to perform a seemingly “simple” task, there can be many barriers. Some of these can be more obvious than others. For example, they may not:

- Be able to type easily.
- Be able to interpret supposedly “obvious” icons.
- Understand keys such as Ctrl or Alt are for.
- Be familiar with what a mouse is or how it is used.
- etc.

If an experiment is being performed to see if a new word processor increases a secretaries proficiency compared to their previous word processor, these types of consideration are likely unwarranted due to the participants expected technological literacy. However, asking elderly people who have not been exposed to computers to use a word processor will have an extremely different result due to a lesser level of *General Technological Literacy*. Any experiment to evaluate such a system must take into account the varying levels of literacy among participants.

Familiarity with Specific Technologies

Even among people with higher levels of *General Technological Literacy*, their ability to use a particular piece of software will depend on their *Familiarity with that Specific Technology*. One of the main examples at the time of writing is the difference between idiomatic Android and iOS user interfaces (UIs). If a user has only ever been exposed to Android UIs, then even the most carefully thought out iOS implementation may result in confusion and thus not be very usable for particular users. The classic problem is that all modern Android devices have a “back” button in the same location, whereas the UX for going “back” in iOS is different. Another, even more specific (but no less important) example is the difference between versions of the *same* OS. Notably, the move from Microsoft Windows 7 to Windows 8 caused much consternation amongst even the most hardened computer users. A similar situation arose when Android moved from Android 4.4 and the “Halo” theme to 5.0 and the “Material” theme.

A naive attempt to eliminate this threat when conducting experiments that use software could be to implement the software for each major platform that participants are expected to be familiar with (provided there is enough resources to overcome the *Artifact Cost* threat). However, then it becomes very difficult to reason about any undesirable effects identified in the experimental results. For example, how would one discern whether such effects were due to:

- Idiosyncrasies between the development process for multiple platforms?
- More technically proficient users gravitating towards one platform in general?
- The underlying design theory being incorrect?

If the differences in experimental results do turn out to be due to idiosyncrasies between the artifact implementation on each platform, then the solution cannot be to make the experience as unified as possible. This would negate the whole point of *Familiarity with Specific Technologies* because each implementation would have to make sacrifices that result in a less idiomatic piece of software.

8.4.3. Addressing Instantiation Validity

Perhaps the best way to address these issues in a research setting is to ensure that appropriate training is provided to each participant before evaluating software. By committing the relevant amount of time to training, the effects of any potentially confusing HCI problems may be reduced. The extent of this practice effect is yet to be determined, but future research should investigate further.

Beyond encouraging participants to practice, it is suggested that at the very least, such threats to IV are identified and accounted for when reporting results of IS evaluations. Further research is required to better understand how the two threats discussed above can be mitigated, as well as identify further threats to IV. Such research will help to ensure that IS research is robust and able to stand by any claims arising from experiments involving software implementations.

8.5. Research Contributions

Despite the evaluations not coming up with ideal results, there were still many contributions made during this thesis. The introductory chapter discussed the expected contributions of this research. They were:

Theoretical Investigating the use of surveys to elicit BNs in place of face to face interviews.

Practical A software tool for constructing BNs via KA.

Methodological Contributions to the theory of DSR with regard to Instantiation Validity.

8.5.1. Contributions to Theory

The first contribution came about due to the lack of theory present in past research using surveys to elicit BNs. In this research, it was shown that surveys can indeed be used as a tool to elicit BNs, but with certain caveats. Unfortunately the BNs elicited during evaluation did not compare favourably to the gold standard network. However, the survey elicitation technique did excel in some areas. It was successfully used to conduct a survey that enlisted a large number of people without a large workload on behalf of the survey administrator. The amount of effort required by the administrator, and by each individual expert, did not change between 5 or 70 people participating. The evaluations did not show any reason why this number couldn't grow arbitrarily high without undue burden on participants or organisers. In addition, the method incorporates ideas from crowd sourcing which were previously foreign to the field of BN elicitation. It is hoped that these ideas can be further developed as research in the field of crowd sourcing continues unabated (see Section 8.7.6, p217).

8.5.2. Contribution to Practice

The practical contribution came in the way of a web application for administering online surveys in order to construct BNs, called BN Elicitator (BNE, Serwylo, 2013). BNE is published as open source software under the GNU GPLv3 License, which encourages contributions to make the tool better. Although the evaluation did not result in a positive result, the licensing of the software ensures that any future work to improve the method is able to build upon the already existing software, rather than starting from scratch.

BNE was used in the evaluation which took place in Chapter 5 and Chapter 7 (car insurance risk assessment). Also, as a proof of concept it has been configured for the completely different domain of water management for wild rivers, though a survey wasn't conducted. The software has two main components, the ability to elicit BN structures, and also to elicit CPTs. Processes that the software takes care of include:

- After configuring BNE with variables of interest, all required questions are automatically generated.
- Participants are automatically allocated questions the first time they log on.
- Automatic collation of responses from multiple experts. This implementation uses the majority vote and ML algorithms, but others could be used in the future.
- Output of the BN to file formats used by major BN software.
- Graphical output of the resulting BN.

The future work in Section 8.7.11 (p220) chapter will discuss plans to further develop BNE to address these and other concerns.

8.5.3. Contributions to Methodology

Section 8.4 discusses additions to Instantiation Validity which were identified during this thesis. In addition to the five threats identified by Lukyanenko et al. (2014), this thesis contributed the two additional threats of *General Technological Literacy* and *Specific Familiarity with Technologies*. After defining the threats, a discussion is presented of how they may be addressed by devoting more time to training users of the system.

8.6. Limitations

The two main limitations of this thesis arose due to the choices taken during evaluation. While the principle of comparing an existing gold standard network to one elicited using a new method under study has been well established (e.g. Kennett et al., 2001; Tsamardinos et al., 2006), there were some limitations due to the specific evaluation undertaken in this thesis. Firstly, the insurance network used as the gold standard (Section 5.1.3) was from a different country, and was over 15 years old by the time the evaluation for this thesis was conducted. Secondly, some assumptions were made when choosing participants for the evaluation in

Chapter 5 and Chapter 7. In hindsight, these assumptions were too strong and the criteria for selecting participants should have been more strict.

The post-hoc analysis described in Section 5.7 investigated whether it was justified to enlist lay participants with experience driving, in preference to insurance experts. The results of that analysis showed that there seems to be a lot of useful information which can be gleamed from using such a population of proxy experts. However, there are also some drawbacks as there will inevitably be questions which would be answered much better by an insurance expert. The results of the evaluation in this thesis showed that BN arcs which have high agreement among the lay participants were quite informative. However, given the post-hoc nature of the analysis which drew this conclusion, further research is required in order to identify how much agreement should be required before deciding to include an arc in the resulting BN.

8.7. Future Work

This research presented the new concept of SEBN, with a large amount of scope for incorporating interesting and useful features to facilitate more effective or efficient elicitation of BNs. However to prevent scope creep, this project focussed on rigorously evaluating an initial proposal to elicit BNs via online surveys. Throughout the thesis, where interesting opportunities for expanding the research were identified, they were documented in this section to maintain focus.

8.7.1. Further Evaluation with Experts

It would be ideal to conduct further evaluations of SEBN with experts. One limitation of the research was that those recruited to take part were not experts, for the reasons discussed in Section 5.1.4 (p106). This means that no inferences can be made about whether experts would have performed better, had they participated. In addition, to provide an even more thorough evaluation, one BN should be elicited using traditional KEBN and a second one via SEBN. A direct comparison of these techniques would provide further insight into when one method is preferable over the other.

8.7.2. Evaluate the BN Elicitator Software

The BNE software was used as a tool to evaluate SEBN presented in Chapter 4 and Chapter 6. What is lacking is a rigorous evaluation of the software itself. In DSR terminology, this equates to an evaluation of an instantiation, rather than using an instantiation to evaluate a method or model. There should be future evaluations to formally evaluate BNE to make sure it adheres to the design in Chapter 4 and Chapter 6. Such evaluations should pay close attention to Instantiation Validity, to ensure there are not any extraneous variables influencing the usability of BNE.

8.7.3. Further Investigation and Validation of the Weighted Sum Algorithm

The weighted sum algorithm from Das (2004) is as follows:

$$\begin{aligned}\Pr(A &= a|X=x, Y=y) \approx \\ &\propto w_x \times \Pr(A=a|CPC(X=x)) + w_y \times \Pr(A=a|CPC(Y=y))\end{aligned}$$

In each case, $\Pr(A = a|CPC(N = n))$ seems to act as a proxy for the simpler $\Pr(A = a|N = n)$. As such, the approximation ends up like so:

$$\begin{aligned}\Pr(A &= a|X=x, Y=y) \approx \\ &\propto w_x \times \Pr(A=a|X=x) + w_y \Pr(A=a|Y=y)\end{aligned}$$

If that is indeed the case, it would be simpler to elicit the conditional probabilities

$\Pr(A = a|N = n)$ directly, rather than the proxy of $\Pr(A = a|CPC(N = n))$. This is discussed by Kim and Pearl (1983) who discuss the principle of approximating $\Pr(A|B, C)$ as proportional to $\Pr(A|B) * \Pr(A|C)$ in certain "dominating" relationships. While Das (2004) does introduce weighting on top of this, they neglect to address the limitations discussed by Kim and Pearl (1983). Further investigation is required to better understand the relationship between CPCs and the conditional probabilities themselves.

Another alternative could be to ask the experts to elicit the top n most compatible parent configurations. Doing so would provide a more accurate estimate of the relevant probabilities, at the expense of more questions and thus more time required of experts.

8.7.4. Variable Elicitation and Classification via Surveys

The beginning of Chapter 4 (p73) mentioned that the scope of this thesis prohibited an approach for eliciting variables through surveys. There are some elicitation techniques which transfer particularly well to the process of survey systems, such as the Repertory Grid (Kelly, 1955), Max100 (Bottomley and Doyle, 2001), or Analytical Hierarchy Process (AHP, Saaty, 1977). These should be investigated to see how they can be used in the elicitation of variables for the subsequent elicitation of a BN structure. Additionally, each variable would also need to have a set of associated states elicited, which are used in the CPT elicitation phase. When considering elicitation of variables and their states, the process of classifying variables should also be considered (Section 4.3, p80). Once elicitation of variables, their states, and their classes are incorporated into SEBN, then it provides an end to end solution for eliciting BNs via online surveys.

8.7.5. Alternate Methods to Elicit BN Structures

Section 4.4 (p85) discussed specifically what questions are allocated to participants for eliciting the structure of BNs. That is, instead of asking several questions of the form "Does X influence Y?", it encouraged asking one question of the form

"Does $\{A, B, C, ...\}$ influence Y?". This allocation of *all parent variables of* Y allows tinkering with the way in which relationships are chosen. Future work could incorporate ideas such as asking "Which n of the following variables influence Y the most?". This will help with restricting the number of parents of a given node, by artificially introducing scarcity resulting in simpler models. It would also ensure there is more information provided by the expert, to be incorporated into crowd sourcing algorithms for collating results.

8.7.6. Investigate Different Collation Algorithms

Dawid and Skene (1979) proposed ML as one algorithm for collating multiple responses together into an authoritative result. It would be worthwhile investigating other, more comprehensive algorithms, to see if they have an effect on the evaluation results. Examples of such are Ipeirotis et al. (2010); Organisciak et al. (2012); Raykar et al. (2010); Sheng et al. (2008); Wauthier and Jordan (2011); Whitehill et al. (2009); Zhou et al. (2012).

In addition to the possibility of better collated BN structures, some of these algorithms provide better mechanisms for inferring the expertise of participants. Section 5.5 (p140) showed that in the evaluation study undertaken, the estimated quality of experts did not correspond well to the actual accuracy they exhibited compared to the gold standard. Although this was only one survey, with one gold standard network, it shows that the Ipeirotis et al. (2010) algorithm may not be suitable in this situation. Other algorithms should also be investigated, such as Bachrach et al. (2012); Wauthier and Jordan (2011), and Raykar et al. (2010).

8.7.7. More Allocations

The BNE software allocates a fixed number of questions to each expert. Sheng et al. (2008) uses active learning to change the proposal from "Each question should be answered by n different experts" to seeking out troublesome questions that seem to divide the experts. Such questions should be allocated to a greater number of experts to facilitate the elicitation of more opinions. For example, experts who have completed the survey could be given the opportunity to continue answering further questions.

This approach helps to better utilise the precious resource that is experts time, directing it to questions which require it more. The main concern with this is that there are multiple phases of elicitation (structure and CPT), and experts who tire themselves out on the first phase may be less willing to return for CPT elicitation.

8.7.8. Heuristics for Removing Cycles

When collating survey responses from multiple experts into BN structures, the result often contains cycles. Section 4.6.1 (p93) discussed one heuristic from Margaritis and Thrun (1999) for deciding which relationships would be the least harmful to remove in order to obtain a DAG. It is worthwhile looking into additional heuristics for deciding which arcs to remove.

Maximizing network score If the CPTs are being learnt from data, allowing CPT elicitation to be conducted by software rather than expert elicitation, then the network score (e.g. BIC) could be maximized, regardless of the strength of the arcs.

Preferring arcs with higher strength Some arcs have stronger support from experts than others (e.g. due to greater majority of experts agreeing, higher self confidence judgements, or an algorithm such as Dawid & Skene). The number of strong arcs removed should be minimized, by preferring to remove weakly supported arcs.

Preferring higher total strength Preferring individual strong arcs may select a small number of strong arcs at the expense of many week arcs. An alternative is to optimize for the highest total strength of all included arcs.

Minimize arc reversals As was shown in the evaluation (Section 5.2.2 p114), the algorithm from Margaritis and Thrun (1999) which includes reversing arcs can result in undesirable reversals and thus non-causal relationships. Optimizing for the minimum number of reversed arcs would reduce this problem.

8.7.9. Modifying BNs

Over time, information about a particular domain may change. In response, it is important to be able to update predictive models that are used in that domain. The technique described in this thesis could be amended to facilitate updating of BNs. One such approach could involve creating mock experts who automatically submit responses to the survey, obtained from knowledge encoded in the previous BN. Change aversion can be introduced by increasing the number of mock experts seeded with previous data. This requires a higher threshold from real expert responses in order to augment the structure. However, modifying the structure of BNs should be done with care, as minor changes to the structure can result in drastic changes to the required CPTs.

8.7.10. Other Odd Looking and Non-Idiomatic Patterns or Optimisations

Various anomalies and opportunities for optimisation were discussed in Section 4.6 (p92). The anomalies discussed include the introduction of cycles and potentially redundant relationships. Opportunities for optimisation included identifying patterns that might indicate NoisyOR and NoisyMAX relationships. Once a framework is in place for presenting anomalies to participants and asking for feedback to help resolve the anomalies, then additional patterns can be searched for. Likewise, additional optimisations could also be searched for.

This is similar in spirit to the concept of BN “idioms” (Neil et al., 2000) or “building blocks” which are reusable patterns that appear in many different BNs. Another similar analogy is software design patterns (Gamma et al., 1994) which help software engineers to re-use solutions to common problems. By understanding when each BN idiom tends to appear, and the type of problem it helps to solve, knowledge engineers are better able to develop new networks.

It should also be noted though, that enlisting the help of survey participants to resolve anomalies or to identify potential for optimisation will increase the burden on them. This goes directly against Proposition 2 (p61). As such, careful thought should be put into whether the improvement to be gained is worth the extra time and effort, especially when compared to using traditional KEBN.

8.7.11. Further Refinement of BNE

Section A.5 (p241) discusses future work to be undertaken to refine the BNE software. This includes changes to the way in which variables are allocated to participants, and moving towards a Software as a Service (SaaS) style of deployment.

Bibliography

Akaike, H. (1998). *Information Theory and an Extension of the Maximum Likelihood Principle*, pages 199–213. Springer Series in Statistics. Springer New York.

Andersen, R., Kasper, J., and Frankel, M. R. (1979). *Total survey error.* Jossey-Bass Publishers.

Arazy, O., Kumar, N., and Shapira, B. (2010). A theory-driven design framework for social recommender systems. *Journal of the Association for Information Systems*, 11(9):455–490.

Armstrong, J. (2001). Combining forecasts. In Armstrong, J., editor, *Principles of Forecasting*, volume 30 of *International Series in Operations Research & Management Science*, pages 417–439. Springer US.

Arnott, D. (2006). Cognitive biases and decision support systems development: a design science approach. *Information Systems Journal*, 16(1):55–78.

Babbie, E. R. (1990). *Survey research methods.* Wadsworth Publishing Company, 2 edition.

Bachrach, Y., Graepel, T., Kasneci, G., Kosinski, M., and Van Gael, J. (2012). Crowd IQ: Aggregating opinions to boost performance. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '12, pages 535–542, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Baker, E., Bosetti, V., Jenni, K. E., and Ricci, E. C. (2014). Facing the experts: Survey mode and expert elicitation. Technical report, Fondazione Eni Enrico Mattei.

Bang-Jensen, J. and Gutin, G. Z. (2009). *Digraphs: Theory, Algorithms and Applications.* Springer Monographs in Mathematics. Springer London, 2nd edition.

Bayes Server Ltd (2015). Bayes Server. https://www.bayesserver.com.

Bazerman, M. H. and Moore, D. A. (2009). *Judgment in managerial decision making*. John Wiley & Sons, 7th edition.

Binder, J., Koller, D., Russell, S., and Kanazawa, K. (1997). Adaptive probabilistic networks with hidden variables. *Machine Learning*, 29:213–244.

Bosnjak, M. and Tuten, T. L. (2003). Prepaid and promised incentives in web surveys: An experiment. *Social Science Computer Review*, 21(2):208–217.

Bottomley, P. A. and Doyle, J. R. (2001). A comparison of three weight elicitation methods: good, better, and best. *Omega*, 29(6):553 – 560.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.

Buchanan, B. G., Davis, R., and Feigenbaum, E. A. (2006). Expert systems: A perspective from computer science. In Ericsson, K. A., Charness, N., Feltovich, P. J., and Hoffman, R. R., editors, *The Cambridge Handbook of Expertise and Expert Performance*, pages 87–104. Cambridge University Press. Cambridge Books Online.

Buntine, W. (1991). Theory refinement on bayesian networks. In *Proceedings of the Seventh conference on Uncertainty in Artificial Intelligence*, UAI'91, pages 52–60, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Buntine, W. (1996). A guide to the literature on learning probabilistic networks from data. *Knowledge and Data Engineering, IEEE Transactions on*, 8(2):195 –210.

Carmines, E. G. and Zeller, R. A. (1979). *Reliability and validity assessment*, volume 17. Sage publications.

Chan, T., McShane, P., and c, H. R. (2011). Uncertainty about uncertainty within a stakeholder group. In *19th International Congress on Modelling and Simulation*.

Chan, T., Ross, H., Hoverman, S., and Powell, B. (2010). Participatory development of a Bayesian network model for catchment-based water resource management. *Water Resources Research*, 46(7).

Chandra, L., Seidel, S., and Gregor, S. (2015). Prescriptive knowledge in is research: Conceptualizing design principles in terms of materiality, action, and boundary conditions. In *Hawaii International Conference on System Sciences*, volume 48.

Chickering, D. M. (2002). Learning equivalence classes of Bayesian-network structures. *J. Mach. Learn. Res.*, 2:445–498.

Chin, K.-S., Tang, D.-W., Yang, J.-B., Wong, S. Y., and Wang, H. (2009). Assessing new product development project risk by Bayesian network with a systematic probability generation methodology. *Expert Systems with Applications*, 36(6):9879–9890.

Chklovski, T. and Gil, Y. (2005). Towards managing knowledge collection from volunteer contributors. In *Proceedings of AAAI Spring Symposium on Knowledge Collection from Volunteer Contributors (KCVC05).*

Clemen, R. T. and Winkler, R. L. (1999). Combining probability distributions from experts in risk analysis. *Risk Analysis*, 19(2):187–203.

Cooper, G. F. (1999). *Computation, Causation, and Discovery*, chapter 1, pages 3–62. The MIT Press.

Cooper, G. F. and Herskovits, E. (1991). A Bayesian method for constructing Bayesian belief networks from databases. In *Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence*, UAI'91, pages 86–94, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Cruz-Ramírez, N., Acosta-Mesa, H.-G., Barrientos-Martínez, R.-E., and Nava-Fernández, L.-A. (2006). How good are the Bayesian information criterion and the minimum description length principle for model selection? a Bayesian network analysis. 4293:494–504.

d'Aquin, M., Motta, E., Sabou, M., Angeletou, S., Gridinoc, L., Lopez, V., and Guidi, D. (2008). Toward a new generation of semantic web applications. *Intelligent Systems, IEEE*, 23(3):20–28.

Das, B. (2004). Generating conditional probabilities for Bayesian networks: Easing the knowledge acquisition problem. Technical report, DSTO.

Dawid, A. P. and Skene, A. M. (1979). Maximum likelihood estimation of observer error-rates using the EM algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):pp. 20–28.

Dieste, O. and Juristo, N. (2011). Systematic review and aggregation of empirical studies on elicitation techniques. *Software Engineering, IEEE Transactions on*, 37(2):283–304.

Dransfeld, H., Pemberton, J., and Jacobs, G. (2000). Quantifying weighted expert opinion: The future of interactive television and retailing. *Technological Forecasting and Social Change*, 63(1):81–90.

Druzdzel, M. and van der Gaag, L. (2000). Building probabilistic networks: "where do the numbers come from?" guest editors' introduction. *Knowledge and Data Engineering, IEEE Transactions on*, 12(4):481–486.

Druzdzel, M. J. and Díez, F. J. (2003). Combining knowledge from different sources in causal probabilistic models. *J. Mach. Learn. Res.*, 4:295–316.

Ericsson, K. A. (2006). An introduction to the cambridge handbook of expertise and expert performance: Its development, organization, and content. In Ericsson, K. A., Charness, N., Feltovich, P. J., and Hoffman, R. R., editors, *The Cambridge Handbook of Expertise and Expert Performance*, pages 3–20. Cambridge University Press. Cambridge Books Online.

Eriksson, H. (1992). A survey of knowledge acquisition techniques and tools and their relationship to software engineering. *Journal of Systems and Software*, 19(1):97 – 107.

Falzon, L. (2006). Using Bayesian network analysis to support centre of gravity analysis in military planning. *European Journal of Operational Research*, 170(2):629 – 643.

Feigenbaum, E. A. (1977). The art of artificial intelligence - themes and case studies of knowledge engineering. Technical report, Standford University Department of Computer Science.

Fenton, N., Neil, M., and Caballero, J. G. (2007). Using ranked nodes to model qualitative judgments in Bayesian networks. *Knowledge and Data Engineering, IEEE Transactions on*, 19(10):1420–1432.

Fernández-López, M., Gómez-Pérez, A., and Juristo, N. (1997). Methontology: From ontological art towards ontological engineering. In *Proceedings of the Ontological Engineering AAAI-97 Spring Symposium Series*.

Fischoff, B. (1981). Debiasing. Technical Report ADA099435, Decision Research.

Flores, M. J., Nicholson, A. E., Brunskill, A., Korb, K. B., and Mascaro, S. (2011). Incorporating expert knowledge when learning Bayesian network structure: A medical case study. *Artificial Intelligence in Medicine*, 53(3):181 – 204.

Free Software Foundation (2007). GNU General Public License, version 3. `http://www.gnu.org/licenses/gpl.html`. Last retrieved 2016-04-01.

Friedman, N., Geiger, D., and Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning*, 29(2-3):131–163.

Friedman, N. and Goldszmidt, M. (1999). Learning in graphical models. chapter Learning Bayesian networks with local structure, pages 421–459. MIT Press, Cambridge, MA, USA.

Friedman, N., Linial, M., Nachman, I., and Pe'er, D. (2000). Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, 7(3-4):601–620.

Gambelli, D. and Bruschi, V. (2010). A Bayesian network to predict the probability of organic farms' exit from the sector: A case study from marche, Italy. *Computers and Electronics in Agriculture*, 71(1):22 – 31.

Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1994). *Design patterns: elements of reusable object-oriented software*. Pearson Education.

Gansner, E. R. and North, S. C. (2000). An open graph visualization system and its applications to software engineering. *Software - Practice and Experience*, 30(11):1203–1233.

Gregor, S. (2006). The nature of theory in information systems. *MIS Quarterly*, 30(3):611 – 642.

Gregor, S. and Hevner, A. R. (2013). Positioning and presenting design science research for maximum impact. *MIS Quarterly*, 37(2):337 – A6.

Gregor, S. and Jones, D. (2007). The anatomy of a design theory. *Journal of the Association for Information Systems*, 8(5):313 – 335.

Groves, R. M. and Lyberg, L. (2010). Total survey error: Past, present, and future. *Public Opinion Quarterly*, 74(5):849–879.

Guilford, J. P. (1978). *Fundamental statistics in psychology and education*. New York : McGraw-Hill, 6 edition.

Hamming, R. W. (1950). Error detecting and error correcting codes. *Bell System Technical Journal*, 29(2):147–160.

Heckerman, D. (1997). Bayesian networks for data mining. *Data Mining and Knowledge Discovery*, 1(1):79–119.

Heckerman, D., Geiger, D., and Chickering, D. M. (1995). Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20:197–243. 10.1007/BF00994016.

Helsper, E. and Van der Gaag, L. (2002). Building Bayesian networks through ontologies. In *ECAI*, pages 680–684.

Henrion, M. (1987). Practical issues in constructing a Bayes' belief network. In *Proceedings of the Third Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-87)*, pages 132–139, New York, NY. Elsevier Science.

Hevner, A., March, S., Park, J., and Ram, S. (2004). Design science in information systems research. *Mis Quarterly*, 28(1):75–105.

Hoffman, R. R. (1998). How can expertise be defined? Implications of research from cognitive psychology. *Exploring expertise*, pages 81–100.

Hoffman, R. R., Shadbolt, N. R., Burton, A., and Klein, G. (1995). Eliciting knowledge from experts: A methodological analysis. *Organizational Behavior and Human Decision Processes*, 62(2):129 – 158.

Hope, L. R., Nicholson, A. E., and Korb, K. B. (2002). Knowledge engineering tools for probability elicitation. Technical report, Monash University.

Hoverman, S., Ross, H., Chan, T., and Powell, B. (2011). Social learning through participatory integrated catchment risk assessment in the Solomon Islands. *Ecology and Society*, 16(2).

Hughes, W. R. (2009). A statistical framework for strategic decision making with AHP: Probability assessment and Bayesian revision. *Omega*, 37(2):463 – 470.

International Organization for Standardization (1985). Information processing – Documentation symbols and conventions for data, program and system flowcharts, program network charts and system resources charts (ISO 5807:1985).

Ipeirotis, P. G., Provost, F., and Wang, J. (2010). Quality management on Amazon mechanical turk. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '10, pages 64–67, New York, NY, USA. ACM.

Jenkinson, D. (2005). The elicitation of probabilities: A review of the statistical literature. Technical report, University of Sheffield, Sheffield, UK.

Jensen, F. V. and Nielsen, T. D. (2007). *Bayesian Networks and Decision Graphs.* Information Science and Statistics. Springer New York.

Johnson, D. B. (1975). Finding all the elementary circuits of a directed graph. *SIAM Journal on Computing*, 4(1):77–84.

Kelly, G. A. (1955). *The psychology of personal constructs. Volume 1: A theory of personality.* WW Norton and Company.

Kennett, R., Korb, K., and Nicholson, A. (2001). Seabreeze prediction using Bayesian networks. In Cheung, D., Williams, G., and Li, Q., editors, *Advances in Knowledge Discovery and Data Mining*, volume 2035 of *Lecture Notes in Computer Science*, pages 148–153. Springer Berlin / Heidelberg. 10.1007/3-540-45357-1_18.

Kim, J. H. and Pearl, J. (1983). A computational model for causal and diagnostic reasoning in inference systems. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence - Volume 1*, IJCAI'83, pages 190–193, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Kittler, J., Hatef, M., Duin, R., and Matas, J. (1998). On combining classifiers. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(3):226–239.

Kjærulff, U. B. and Madsen, A. L. (2008). *Bayesian Networks and Influence*

Diagrams: A Guide to Construction and Analysis. Springer. ISBN: 978-0-387-74100-0.

Kjærulff, U. B. and Madsen, A. L. (2013). *Bayesian Networks and Influence Diagrams: A Guide to Construction and Analysis*, volume 22 of *Information Science and Statistics.* Springer, 2nd edition.

Kohavi, R. and Provost, F. (1998). Glossary of terms. *Machine Learning*, 30(2-3):271–274.

Korb, K. and Nicholson, A. (2011). *Bayesian artificial intelligence.* CRC Press, 2nd edition.

Krueger, T., Page, T., Hubacek, K., Smith, L., and Hiscock, K. (2012). The role of expert opinion in environmental modelling. *Environmental Modelling & Software*, 36:4–18.

Kuikka, S. and Varis, O. (1997). Uncertainties of climatic change impacts in finnish watersheds : a Bayesian network analysis of expert knowledge. (1):128.

Lam, L. and Suen, C. (1994). A theoretical analysis of the application of majority voting to pattern recognition. In *Pattern Recognition, 1994. Vol. 2 - Conference B: Computer Vision amp; Image Processing., Proceedings of the 12th IAPR International. Conference on*, volume 2, pages 418–420 vol.2.

Larrick, R. P. (2004). Debiasing. In Koehler, D. J. and Harvey, N., editors, *Blackwell Handbook of Judgment and Decision Making.* Blackwell Publishing.

Laskey, K. B. and Mahoney, S. M. (1997). Network fragments: representing knowledge for constructing probabilistic models. In *Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence*, UAI'97, pages 334–341, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Lichtenstein, S., Slovic, P., Fischhoff, B., Layman, M., and Combs, B. (1978). Judged frequency of lethal events. *Journal of Experimental Psychology: Human Learning and Memory*, 4(6):551 – 578.

Lucas, P. J., van der Gaag, L. C., and Abu-Hanna, A. (2004). Bayesian networks in biomedicine and health-care. *Artificial Intelligence in Medicine*, 30(3):201–214.

Lukyanenko, R., Evermann, J., and Parsons, J. (2014). Instantiation validity in is design research. In Tremblay, M., Chiarini, Vander, Meer, D., Rothenberger, M., Gupta, A., and Yoon, V., editors, *Advancing the Impact of Design Science: Moving from Theory to Practice*, volume 8463 of *Lecture Notes in Computer Science*, pages 321–328. Springer International Publishing.

Lukyanenko, R., Evermann, J., and Parsons, J. (2015). Guidelines for establishing instantiation validity in IT artifacts: A survey of IS research. In Donnellan, B., Helfert, M., Kenneally, J., VanderMeer, D., Rothenberger, M., and Winter, R., editors, *New Horizons in Design Science: Broadening the Research Agenda*, volume 9073 of *Lecture Notes in Computer Science*, pages 430–438. Springer International Publishing.

Lukyanenko, R. and Parsons, J. (2013). Reconciling theories with design choices in design science research. In vom Brocke, J., Hekkala, R., Ram, S., and Rossi, M., editors, *Design Science at the Intersection of Physical and Virtual Design*, volume 7939 of *Lecture Notes in Computer Science*, pages 165–180. Springer Berlin Heidelberg.

Mahoney, S. M. and Laskey, K. B. (1996). Network engineering for complex belief networks. In *Proceedings of the Twelfth International Conference on Uncertainty in Artificial Intelligence*, UAI'96, pages 389–396, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

March, S. T. and Smith, G. F. (1995). Design and natural science research on information technology. *Decision Support Systems*, 15(4):251 – 266.

Marcus, B., Bosnjak, M., Lindner, S., Pilischenko, S., and Schütz, A. (2007). Compensating for low topic interest and long surveys: A field experiment on nonresponse in web surveys. *Social Science Computer Review*, 25(3):372–383.

Margaritis, D. and Thrun, S. (1999). Bayesian Network Induction via Local Neighborhoods. In Solla, S., Leen, T., and Müller, K.-R., editors, *Proceedings of Conference on Neural Information Processing Systems (NIPS-12)*. MIT Press.

Martin, T. G., Burgman, M. A., Fidler, F., Kuhnert, P. M., Low-Choy, S., McBride, M., and Mengersen, K. (2012). Eliciting expert knowledge in conservation science. *Conservation Biology*, 26(1):29–38.

Martin, T. G., Kuhnert, P. M., Mengersen, K., and Possingham, H. P. (2005). The power of expert opinion in ecological models using Bayesian methods: Impact of grazing on birds. *Ecological Applications*, 15(1):266–280.

Martínez, I., Moral, S., Rodríguez, C., and Salmerón, A. (2002). Factorisation of probability trees and its application to inference in Bayesian networks. In Gámez, J. and Salmerón, A., editors, *Procs. of the 1st European Workshop on Probabilistic Graphical Models*, pages 127–134.

Mc Culloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.

McGraw, K. and Seale, M. (1988). Knowledge elicitation with multiple experts: considerations and techniques. *Artificial Intelligence Review*, 2(1):31–44.

Milton, N. R. (2008). *Knowledge Acquisition in Practice: A Step by Step Guide*. Springer.

Monti, S. and Carenini, G. (2000). Dealing with the expert inconsistency in probability elicitation. *Knowledge and Data Engineering, IEEE Transactions on*, 12(4):499–508.

Naveh, B. and Contributors (2015). JGraphT. http://jgrapht.org/. Last retrieved 2016-04-01.

Neil, M., Fenton, N., and Nielson, L. (2000). Building large-scale Bayesian networks. *Knowl. Eng. Rev.*, 15:257–284.

Neyman, J. (1934). On the two different aspects of the representative method: The method of stratified sampling and the method of purposive selection. *Journal of the Royal Statistical Society*, 97(4):pp. 558–625.

Nicholson, A. E., Twardy, C. R., Korb, K. B., and Hope, L. R. (2008). *Decision Support for Clinical Cardiovascular Risk Assessment*, chapter 3, pages 33–52. John Wiley & Sons, Ltd.

Nickerson, R. S. (1998). Confirmation bias: A ubiquitous phenomenon in many guises. *Review of General Psychology*, 2(2):175–220.

Norsys Software Corp (2016). Netica. https://www.norsys.com/. Last retrieved 2016-04-01.

Nunamaker Jr, J. and Chen, M. (1990). Systems development in information systems research. *Proceedings of the Twenty-Third Annual Hawaii International Conference on System Sciences*, 3.

O'Leary, R., Fisher, R., Low-Choy, S., Mengersen, K., and Caley, M. (2011). What is an expert? In *19th International Congress on Modelling and Simulation. Modelling and Simulation Society of Australia and New Zealand*, volume 9, pages 2149–2155.

Olesen, K. G., Kjærulff, U. B., Jensen, F., Jensen, F. V., Falck, B., Andreassen, S., and Andersen, S. K. (1989). A munin network for the median nerve - a case study on loops. *Applied Artificial Intelligence*, 3(2-3):385–403.

Oniško, A. (2008). *Medical Diagnosis*, chapter 2, pages 15–32. John Wiley & Sons, Ltd.

Onwuegbuzie, A. J., Dickinson, W. B., Leech, N. L., and Zoran, A. G. (2009). A qualitative framework for collecting and analyzing data in focus group research. *International Journal of Qualitative Methods*, 8(3):1–21.

Organisciak, P., Efron, M., Fenlon, K., and Senseney, M. (2012). Evaluating rater quality and rating difficulty in online annotation activities. *Proceedings of the American Society for Information Science and Technology*, 49(1):1–10.

Oxford (2016). *Oxford English Dictionary*. Oxford University Press.

Pearl, J. (1986). Fusion, propagation, and structuring in belief networks. *Artificial Intelligence*, 29(3):241 – 288.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Peffers, K., Tuunanen, T., Rothenberger, M., and Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3):45–77.

Pellikka, J., Kuikka, S., Lindén, H., and Varis, O. (2005). The role of game management in wildlife populations: uncertainty analysis of expert knowledge. *European Journal of Wildlife Research*, 51(1):48–59.

Pollino, C. A., Woodberry, O., Nicholson, A., Korb, K., and Hart, B. T. (2007). Parameterisation and evaluation of a Bayesian network for use in an ecological risk assessment. *Environmental Modelling & Software*, 22(8):1140 – 1152.

Pradhan, M., Provan, G., Middleton, B., and Henrion, M. (1994). Knowledge engineering for large belief networks. In *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, UAI'94, pages 484–490, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Price, P. C. (1998). Effects of a relative-frequency elicitation question on likelihood judgment accuracy: The case of external correspondence. *Organizational Behavior and Human Decision Processes*, 76(3):277 – 297.

Pries-Heje, J., Baskerville, R., Richard, and Venable, J. R. (2008). Strategies for design science research evaluation. In *ECIS 2008 Proceedings*.

Project Troia (2013). Troia server. https://github.com/ipeirotis/Troia-Server. Last retrieved 2016-04-01.

Przytula, K. and Thompson, D. (2000). Construction of Bayesian networks for diagnostics. In *Aerospace Conference Proceedings, 2000 IEEE*, volume 5, pages 193–200.

Quinn, A. J. and Bederson, B. B. (2011). Human computation: A survey and taxonomy of a growing field. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1403–1412, New York, NY, USA. ACM.

R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Raykar, V. C., Yu, S., Zhao, L. H., Valadez, G. H., Florin, C., Bogoni, L., and Moy, L. (2010). Learning from crowds. *Journal of Machine Learning Research*, 11:1297–1322.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, 14(5):465 – 471.

Rumantir, G. (2003). A Bayesian framework for tropical cyclone causal and risk analysis. In *Second International Conference on Knowledge Economy and Development of Science and Technology.*

Russell, S. J. and Norvig, P. (2010). *Artificial intelligence: a modern approach*, volume 3. Upper Saddle River, N.J. : Prentice Hall.

Saaty, T. L. (1977). A scaling method for priorities in hierarchical structures. *Journal of Mathematical Psychology*, 15(3):234 – 281.

Saaty, T. L. (1990). How to make a decision: The analytic hierarchy process. *European Journal of Operational Research*, 48(1):9–26.

Schafer, J. and Graham, J. W. (2002). Missing data: Our view of the state of the art. *Psychological Methods*, 7(2):147–177.

Schapire, R. (1990). The strength of weak learnability. *Machine Learning*, 5(2):197–227.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464.

Scutari, M. (2010). Learning Bayesian Networks with the bnlearn R Package. *Journal of Statistical Software*, 35(3):1–22.

Seni, G. and Elder, J. F. (2010). Ensemble methods in data mining: Improving accuracy through combining predictions. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 2(1):1–126.

Serwylo, P. (2013). BN Elicitator. https://github.com/bn-elicitator/bn-elicitator. Last retrieved 2016-04-01.

Sheng, V. S., Provost, F., and Ipeirotis, P. G. (2008). Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 614–622, New York, NY, USA. ACM.

Shenton, W., Hart, B. T., and Chan, T. U. (2013). A Bayesian network approach to support environmental flow restoration decisions in the Yarra river, Australia. *Stochastic Environmental Research and Risk Assessment*, pages 1–9.

Shortliffe, E. H., Scott, A. C., Bischoff, M. B., Campbell, A. B., Van Melle, W., and Jacobs, C. D. (1984). *An expert system for oncology protocol management*, chapter 35, pages 653–665. Addison-Wesley.

Simon, H. (1969). *The sciences of the artificial.* The MIT Press, 1st edition.

Simon, H. (1996). *The sciences of the artificial.* The MIT Press, 3rd edition.

Sniezek, J. A. and Henry, R. A. (1989). Accuracy and confidence in group judgment. *Organizational Behavior and Human Decision Processes*, 43(1):1 – 28.

Snow, R., O'Connor, B., Jurafsky, D., and Ng, A. Y. (2008). Cheap and fast–but is it good?: Evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 254–263, Stroudsburg, PA, USA. Association for Computational Linguistics.

Spiegelhalter, D. J. and Lauritzen, S. L. (1990). Sequential updating of conditional probabilities on directed graphical structures. *Networks*, 20(5):579–605.

Spirtes, P., Glymour, C. N., and Scheines, R. (2000). *Causation, Prediction, and Search.* Adaptive Computation and Machine Learning. MIT Press.

Studer, R., Benjamins, V., and Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 25:161 – 197.

Tarjan, R. E. (1973). Enumeration of the elementary circuits of a directed graph. *SIAM Journal on Computing*, 2(3):211–216.

Tiernan, J. C. (1970). An efficient search algorithm to find the elementary circuits of a graph. *Communications of the Association for Information Systems*, 13(12):722–726.

Tsamardinos, I., Brown, L. E., and Aliferis, C. F. (2006). The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65(1):31–78.

Tversky, A. and Kahneman, D. (1973). Availability: A heuristic for judging frequency and probability. *Cognitive Psychology*, 5(2):207 – 232.

Tversky, A. and Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157):1124–1131.

Van Aken, J. E. (2005). Management research as a design science: Articulating the research products of mode 2 knowledge production in management. *British Journal of Management*, 16(1):19–36.

van der Gaag, L., Renooij, S., Witteman, C., Aleman, B., and Taal, B. (1999). How to elicit many probabilities. In *Proceedings of the Fifteenth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)*, pages 647–654, San Francisco, CA.

van der Gaag, L., Renooij, S., Witteman, C., Aleman, B., and Taal, B. (2002). Probabilities for a probabilistic network: a case study in oesophageal cancer. *Artificial Intelligence in Medicine*, 25(2):123 – 148.

van der Gaag, L. C. and Helsper, E. M. (2002). Experiences with modelling issues in Building probabilistic networks. In Gómez-Pérez, A. and Benjamins, V., editors, *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, volume 2473 of *Lecture Notes in Computer Science*, pages 21–26. Springer Berlin Heidelberg.

Venable, J., Pries-Heje, J., and Baskerville, R. (2012). A comprehensive framework for evaluation in design science research. In Peffers, K., Rothenberger, M., and Kuechler, B., editors, *Design Science Research in Information Systems. Advances in Theory and Practice*, volume 7286 of *Lecture Notes in Computer Science*, pages 423–438. Springer Berlin Heidelberg.

Wagenmakers, E.-J., Wetzels, R., Borsboom, D., van der Maas, H. L. J., and Kievit, R. A. (2012). An agenda for purely confirmatory research. *Perspectives on Psychological Science*, 7(6):632–638.

Wallace, C. S. and Korb, K. B. (1999). Learning linear causal models by mml sampling. In Gammerman, A., editor, *Causal Models and Intelligent Data Management*, pages 89–111. Springer Berlin Heidelberg.

Walls, J. G., Widermeyer, G. R., and El Sawy, O. A. (2004). Assessing information system design theory in perspective: how useful was our 1992 initial rendition? *Journal of Information Technology Theory and Application (JITTA)*, 6(2):43–58.

Walls, J. G., Widmeyer, G. R., and El Sawy, O. A. (1992). Building an information system design theory for vigilant EIS. *Information Systems Research*, 3(1):36–59.

Wauthier, F. L. and Jordan, M. I. (2011). Bayesian bias mitigation for crowdsourcing. In *Proc. of NIPS*.

Weisberg, H. F. (2009). *The total survey error approach: A guide to the new science of survey research*. University of Chicago Press.

Whitehill, J., Ruvolo, P., Wu, T., Bergsma, J., and Movellan, J. R. (2009). Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *NIPS*, volume 22, pages 2035–2043.

Wintre, M. G., North, C., and Sugar, L. A. (2001). Psychologists' response to criticisms about research based on undergraduate participants: A developmental perspective. *Canadian Psychology*, 42(3):216–225.

wu Liao, X., Wan, T., and Li, Y. (2008). A Bayesian network model under group decision making for evaluating IT outsourcing risk. In *The 2008 International Conference on Risk Management & Engineering Management*, pages 559–564, Los Alamitos, CA, USA. IEEE Computer Society.

Xiao-xuan, H., Hui, W., and Shuo, W. (2007). Using expert's knowledge to build Bayesian networks. In *International Conference on Computational Intelligence and Security Workshops, 2007*, pages 220 –223.

Yager, R. R. (1979). An eigenvalue method of obtaining subjective probabilities. *Behavioral Science*, 24(6):382–387.

Zhang, Y., Hu, Q., Zhang, W., and Liu, J. (2012). A novel bayesian network structure learning algorithm based on maximal information coefficient. In *Advanced Computational Intelligence (ICACI), 2012 IEEE Fifth International Conference on*, pages 862–867.

Zhou, D., Basu, S., Mao, Y., and Platt, J. C. (2012). Learning from the wisdom of crowds by minimax entropy. In Pereira, F., Burges, C., Bottou, L., and Weinberger, K., editors, *Advances in Neural Information Processing Systems 25*, pages 2195–2203. Curran Associates, Inc.

A. BNE Implementation Details

This appendix presents details of the BNE software (Serwylo, 2013) which implements SEBN, both for structure and CPT elicitation. The software is licensed under the GNU GPLv3 license (Free Software Foundation, 2007). BNE is a web application, and was used as the survey software for evaluating SEBN in Chapter 5 and 7. The evaluations were conducted by configuring BNE with variables from the car insurance network (Binder et al., 1997), although the software is built in such a way as to be agnostic of the specific BN which is being elicited.

The following section will discuss the technical aspects of the software, before Section A.2 provides an overview of how to configure the system. Section A.3 then discusses some specifics of how BNE was used to conduct SEBN in this thesis. Analysis of survey results using BNE is discussed in Section A.4, before Section A.5 discusses avenues for future research and development involving BNE.

A.1. Tech Stack

The software is written in Groovy[1] which is a JVM based language. Groovy allows writing code which ineroperates with existing Java libraries, but using a more concise syntax than Java. It also provided many functional features which eased development before they were implemented and released in Java 8.

The framework used to build the application was Grails[2]. Grails was chosen as it focuses on "convention over configuration" which emphasises sane configuration

[1] `http://groovy-lang.org` (Last retrieved 2016-04-01)

[2] `https://grails.org` (Last retrieved 2016-04-01)

default. This resulted in less time configuring the Grails framework and more time developing BNE.

Grails applications can use most popular database engines. For the purpose of the evaluations in this thesis, MySQL[3] was used due to the authors familiarity with it.

The Grails framework outputs J2EE applications, and BNE is no exception. BNE can be deployed in any J2EE container software. For the purpose of the evaluations in this thesis Apache Tomcat[4] was chosen as it was available in the official Debian repositories. The Tomcat server sat behind an Apache2[5] reverse proxy.

A.2. Configuration

In order to configure the system, a configuration file must be provided. The one used by the evaluations in this thesis is available at `https://github.com/bn-elicitator/bn-elicitator/blob/v0.0.3/grails-app/utils/bn/elicitator/init/loaders/InsuranceDataLoader.groovy`.

First and foremost, the configuration file must specify the list of variables. This is because SEBN does not specify how to elicit variables. Rather, at this point they must be specified from some predetermined source (e.g. traditional KEBN). When specifying variables, detailed descriptions are also required so they can be shown as contextual help whenever a variable is displayed to the user. In addition, the way in which the variable should be phrased when put into a procedurally generated question is configured.

Once the variables are specified, they are then classified. The default implementation of the configuration file allows for background, mediating, problem, and symptom variables as per Kjærulff and Madsen (2013). These are the classes used for the insurance network during the evaluations. However, `https://github.com/bn-elicitator/bn-elicitator/blob/v0.0.3/grails-app/utils/bn/elicitator/`

[3] `https://dev.mysql.com` (Last retrieved 2016-04-01)

[4] `https://tomcat.apache.org` (Last retrieved 2016-04-01)

[5] `https://httpd.apache.org` (Last retrieved 2016-04-01)

`init/loaders/Chan2010DataLoader.groovy` shows an example configuration file for the Chan et al. (2010) water management network. In this case, the constraints between each variable class are also specified. The default configuration is setup with the generic variable classes with dependencies shown in Figure 4.7 (p83).

Finally, the configuration file must specify the states each variable can take. As with the variables themselves, it is important to also provide detailed labels for each state, so that it can be used to generate questions as per Section 6.2 (p156).

A.3. Conducting SEBN via BNE

Experts are either able to signup for the system themselves (using their Facebook account or creating an account in BNE), or be manually added via the administration interface.

When running the system, there are two modes:

1. Structure elicitation
2. Probability elicitation

The structure elicitation mode is an implementation of SEBN as per Chapter 4, while the probability elicitation follows the process in Chapter 6.

The first time participants sign in they are allocated a set of questions as per Section 4.4 (p85) or Section 6.3 (p168), depending on the mode. The configuration file allows the administrator to specify how many questions should be allocated to each participant.

As participants work their way through their allocated survey questions, BNE logs information such as when they started the survey, when each question is answered, and when they finish their survey. This is so that BNE can be used as a research tool to investigate SEBN. Of particular interest to this thesis was how much time is required of experts to elicit BNs.

Another facet of BNE which makes it useful as a tool for researching SEBN is the option of conducting questionnaires at the conclusion of the probability elicitation survey. This feedback survey is separate from the SEBN, and exists for the purpose of collecting feedback about users experience using BNE and SEBN. During the evaluation survey in Chapter 7, this was also used in order to collect demographic information about the participants. Administrators are able to configure arbitrary questions if they choose to use BNE for their own studies.

A.4. Analysing Results

BNE provides several different ways to analyse the BN structures and CPTs elicited during surveys. These features were implemented primarily so that the software could be used as a tool to evaluate SEBN. Much of the analysis done in the evaluation chapters of this thesis made use of BNE to calculate the required metrics.

Firstly, BNE is capable of collating several different BN structures using different majority vote thresholds, or EM priors. This results in several different candidate structures for comparison. It will also estimate the quality of each expert when collating structures using EM.

If provided with a gold standard network, as was the case in the evaluations performed during this thesis, then BNE will calculate the following metrics for each collated structure:

- Number of arcs
- SHD
- ROC metrics (i.e. TPR, FPR)

Other analysis performed by BNE for the purpose of evaluation during this thesis includes attempting to verify the intuitions from the Das (2004) algorithm. This is done by comparing explicitly elicited conditional probabilities with those inferred from the weighted sum algorithm.

A.5. Future work

A.5.1. GUI Configuration

Some of the configuration in BNE can be achieved via a GUI. Examples include adding new user accounts for experts to sign in and editing the landing page. However, even more of the configuration is done via text files. Examples include adding and classifying variables. By allowing all of BNE to be configured via the GUI, users needn't have access to the server in order to configure it.

A.5.2. Multiple surveys on one installation

Currently, the software implementation of SEBN is quite general, and able to be used for construction of other BNs beyond those created for the evaluations in this thesis. The caveat is that setting up BNE to administer surveys is quite a manual process. With an automated deployment process and a further refined user interface, this process can be made easier. The end result is that no technical skills would be required to initiate a survey and thus construct a BN.

Improving the software such that lay users are able to make use of it should also allow it to be deployed using a Software as a Service (Saas) model. Most current software for eliciting BNs is still sold as installable programs. SaaS would allow people wishing to utilise BNs to log on to a website, enter the data they are interested in eliciting, then sending links to experts asking them to complete the survey. The output of these completed surveys would be a complete BN, along with metrics such as how confident BNE is in certain relationships.

A.5.3. Allocating Variables Based on Number of Responses

The implementation in the BNE software built for this research chooses questions to allocate based on those which had been *allocated* to the fewest experts. However, this should have been questions which hadn't been *answered* yet. During the evaluation in Chapter 5, some variables were only answered by 2 people despite 6 being allocated. This would not have been problematic if the allocation strategy favoured questions answered the least number of times, as a variable answered by 2 people would be at the front of the queue to be allocated to another participant.

B. Results of Collating Responses

In order to produce a valid BN structure from SEBN, the collated survey responses must form an acyclic graph. Section 5.2.2 (p114) discussed the Margaritis and Thrun (1999) cycle removal algorithm, to ensure an acyclic graph for use in a BN. The following sections show the result of applying this algorithm to the *Survey* structures from the evaluation in Chapter 5. At each stage of the algorithm, the total number of arcs in the network are shown, as is the arc which appears in the most of these and thus is reversed before proceeding to the next iteration.

B.1. Cycle Removal for Maj Network Structures

The $Maj_{4,5,6}$ network structures all had zero arcs removed. Thus, this section shows the arcs that were reversed in the Maj_3 and Maj_2 structures in order to ensure they were acyclic.

Cycles in Maj_3

Iteration	Total cycles	Worst arc	Appeared in (cycles)
1	76	Accident → DrivHist	40
2	36	Accident→ SeniorTrain	25
3	11	SeniorTrain→ DrivHist	4
4	7	OtherCarCost→ Age	3
5	4	DrivHist→ DrivQuality	2
6	2	Theft→ AntiTheft	1
7	1	DrivHist→ DrivingSkill	1

Table B.1.: Variables removed to eliminate cycles in the Maj_3 network structures.

Cycles in Maj_2

Iteration	Total cycles	Worst arc	Appeared in (cycles)
1	6569858	SeniorTrain→ RiskAversion	4148803
2	2421055	OtherCarCost→ Age	1414201
3	1006854	OtherCarCost→ DrivingSkill	679988
4	326866	ThisCarCost→ Age	166567
5	160299	DrivingSkill→ Age	98183
6	62116	ThisCarCost→ SocioEcon	37765
7	24653	ThisCarCost→ MakeModel	18110
8	6543	RuggedAuto→ MakeModel	4094
9	2449	DrivHist→ Theft	2138
10	311	DrivHist→ DrivQuality	103
11	208	DrivingSkill→ DrivQuality	68
12	140	HomeBase→ SocioEcon	28
13	118	Theft→ MakeModel	27
14	91	DrivHist→ DrivingSkill	26
15	65	DrivHist→ SeniorTrain	18
16	47	Accident→ SeniorTrain	18
17	29	Airbag→ VehicleYear	5
18	24	Airbag→ MakeModel	5
19	19	ThisCarDam→ SeniorTrain	5
20	14	Antilock→ VehicleYear	3
21	11	AntiTheft→ Theft	2
22	9	MakeModel→ SocioEcon	2
23	7	OtherCar→ DrivingSkill	2
24	5	Theft→ HomeBase	1
25	4	VehicleYear→ Mileage	1
26	3	SeniorTrain→ DrivingSkill	1
27	2	DrivHist→ Accident	1
28	1	RuggedAuto→ ThisCarDam	1

Table B.2.: Variables removed to eliminate cycles in the Maj_2 network structures.

B.2. Cycle Removal for DS Network Structures

Cycles in $DS_{0.001}$

Iteration	Total cycles	Worst arc	Appeared in (cycles)
1	24282	ThisCarCost → Age	16095
2	8187	ThisCarCost → MakeModel	6895
3	1292	DrivHist → DrivQuality	647
4	645	DrivHist → DrivingSkill	421
5	224	OtherCar → DrivingSkill	169
6	55	SeniorTrain → RiskAversion	37
7	18	DrivHist → SeniorTrain	5
8	13	Accident → SeniorTrain	3
9	10	DrivQuality → DrivingSkill	2
10	8	Theft → HomeBase	2
11	6	Airbag → MakeModel	2
12	4	Theft → AntiTheft	1
13	3	Age → DrivingSkill	1
14	2	Airbag → VehicleYear	1
15	1	VehicleYear → Antilock	1

Table B.3.: Variables removed to eliminate cycles in the $DS_{0.001}$ network structure.

Cycles in $DS_{0.01,0.05}$

The $DS_{0.01,0.05}$ network structures resulted in the same output from the Margaritis and Thrun (1999) algorithm.

Iteration	Total cycles	Worst arc	Appeared in (cycles)
1	27212	ThisCarCost→ Age	16187
2	11025	ThisCarCost→ MakeModel	8913
3	2112	DrivHist→ DrivQuality	1057
4	1055	DrivHist→ DrivingSkill	657
5	398	OtherCar→ DrivingSkill	267
6	131	SeniorTrain→ RiskAversion	109
7	22	DrivHist→ SeniorTrain	5
8	17	RiskAversion→ Age	4
9	13	Accident→ SeniorTrain	3
10	10	DrivQuality→ DrivingSkill	2
11	8	Theft→ HomeBase	2
12	6	Airbag→ MakeModel	2
13	4	Theft→ AntiTheft	1
14	3	Age→ DrivingSkill	1
15	2	Airbag→ VehicleYear	1
16	1	VehicleYear→ Antilock	1

Table B.4.: Variables removed to eliminate cycles in the $DS_{0.01,0.05}$ network structures.

Cycles in $DS_{0.10}$

Iteration	Total cycles	Worst arc	Appeared in (cycles)
1	54469	ThisCarCost→ Age	34654
2	19815	ThisCarCost→ MakeModel	15386
3	4429	Accident→ SeniorTrain	2161
4	2268	DrivHist→ DrivQuality	1077
5	1191	DrivHist→ DrivingSkill	581
6	610	DrivHist→ SeniorTrain	527
7	83	Airbag→ DrivingSkill	50
8	33	OtherCar→ DrivingSkill	12
9	21	RiskAversion→ Age	8
10	13	Airbag→ MakeModel	3
11	10	Theft→ HomeBase	2
12	8	DrivQuality→ DrivingSkill	2
13	6	Theft→ AntiTheft	1
14	5	Age→ DrivingSkill	1
15	4	RiskAversion→ SeniorTrain	1
16	3	Airbag→ VehicleYear	1
17	2	VehicleYear→ Antilock	1
18	1	DrivHist→ Accident	1

Table B.5.: Variables removed to eliminate cycles in the $DS_{0.10}$ network structure.

Cycles in $DS_{0.15}$

Iteration	Total cycles	Worst arc	Appeared in (cycles)
1	350757	ThisCarCost→ MakeModel	176762
2	173995	ThisCarCost→ Age	130887
3	43108	OtherCarCost→ Age	23924
4	19287	OtherCarCost→ DrivingSkill	10617
5	8670	SeniorTrain→ RiskAversion	3637
6	5033	DrivQuality→ RiskAversion	2406
7	2659	DrivingSkill→ RiskAversion	1702
8	957	DrivingSkill→ Age	730
9	227	Accident→ DrivQuality	81
10	146	DrivHist→ DrivQuality	60
11	86	DrivHist→ DrivingSkill	42
12	44	SeniorTrain→ OtherCar	24
13	20	DrivHist→ SeniorTrain	4
14	16	Accident→ SeniorTrain	4
15	12	Airbag→ MakeModel	3
16	9	Theft→ HomeBase	2
17	7	OtherCar→ DrivingSkill	2
18	5	Theft→ AntiTheft	1
19	4	DrivQuality→ DrivingSkill	1
20	3	Airbag→ VehicleYear	1
21	2	VehicleYear→ Antilock	1
22	1	DrivHist→ Accident	1

Table B.6.: Variables removed to eliminate cycles in the $DS_{0.15}$ network structure.

Cycles in $DS_{0.20}$

Iteration	Total cycles	Worst arc	Appeared in (cycles)
1	663271	ThisCarCost→ Age	279515
2	383756	ThisCarCost→ MakeModel	233892
3	149864	Theft→ MakeModel	93959
4	55905	OtherCarCost→ Age	31811
5	24197	OtherCarCost→ DrivingSkill	13390
6	10807	SeniorTrain→ RiskAversion	4560
7	6247	DrivQuality→ RiskAversion	2998
8	3281	DrivingSkill→ RiskAversion	2111
9	1170	DrivingSkill→ Age	943
10	227	Accident→ DrivQuality	81
11	146	DrivHist→ DrivQuality	60
12	86	DrivHist→ DrivingSkill	42
13	44	SeniorTrain→ OtherCar	24
14	20	DrivHist→ SeniorTrain	4
15	16	Accident→ SeniorTrain	4
16	12	Airbag→ MakeModel	3
17	9	Theft→ HomeBase	2
18	7	OtherCar→ DrivingSkill	2
19	5	Theft→ AntiTheft	1
20	4	DrivQuality→ DrivingSkill	1
21	3	Airbag→ VehicleYear	1
22	2	VehicleYear→ Antilock	1
23	1	DrivHist→ Accident	1

Table B.7.: Variables removed to eliminate cycles in the $DS_{0.20}$ network structure.

Cycles in $DS_{0.25}$

Iteration	Total cycles	Worst arc	Appeared in (cycles)
1	812194	ThisCarCost→ MakeModel	314153
2	498041	ThisCarCost→ Age	279515
3	218526	Theft→ MakeModel	107986
4	110540	HomeBase→ Age	54635
5	55905	OtherCarCost→ Age	31811
6	24197	OtherCarCost→ DrivingSkill	13390
7	10807	SeniorTrain→ RiskAversion	4560
8	6247	DrivQuality→ RiskAversion	2998
9	3281	DrivingSkill→ RiskAversion	2111
10	1170	DrivingSkill→ Age	943
11	227	Accident→ DrivQuality	81
12	146	DrivHist→ DrivQuality	60
13	86	DrivHist→ DrivingSkill	42
14	44	SeniorTrain→ OtherCar	24
15	20	DrivHist→ SeniorTrain	4
16	16	Accident→ SeniorTrain	4
17	12	Airbag→ MakeModel	3
18	9	Theft→ HomeBase	2
19	7	OtherCar→ DrivingSkill	2
20	5	Theft→ AntiTheft	1
21	4	DrivQuality→ DrivingSkill	1
22	3	Airbag→ VehicleYear	1
23	2	VehicleYear→ Antilock	1
24	1	DrivHist→ Accident	1

Table B.8.: Variables removed to eliminate cycles in the $DS_{0.25}$ network structure.

Cycles in $DS_{0.30}$

Iteration	Total cycles	Worst arc	Appeared in (cycles)
1	1776719	ThisCarCost→ Age	655710
2	1121009	ThisCarCost→ MakeModel	532717
3	588292	AntiTheft→ RiskAversion	243032
4	345260	HomeBase→ Age	135872
5	242602	Theft→ MakeModel	143931
6	98671	OtherCarCost→ Age	56039
7	43043	OtherCarCost→ DrivingSkill	21908
8	21135	RuggedAuto→ MakeModel	9080
9	12055	SeniorTrain→ RiskAversion	5342
10	6713	DrivQuality→ RiskAversion	3394
11	3351	DrivingSkill→ RiskAversion	2157
12	1194	DrivingSkill→ Age	966
13	228	Accident→ DrivQuality	81
14	147	DrivHist→ DrivQuality	60
15	87	DrivHist→ DrivingSkill	42
16	45	SeniorTrain→ OtherCar	24
17	21	DrivHist→ SeniorTrain	4
18	17	Accident→ SeniorTrain	4
19	13	Airbag→ MakeModel	3
20	10	Theft→ HomeBase	2
21	8	OtherCar→ DrivingSkill	2
22	6	Theft→ AntiTheft	1
23	5	DrivQuality→ DrivingSkill	1
24	4	Airbag→ VehicleYear	1
25	3	VehicleYear→ Antilock	1
26	2	DrivHist→ Accident	1
27	1	RuggedAuto→ ThisCarDam	1

Table B.9.: Variables removed to eliminate cycles in the $DS_{0.30}$ network structure.

Cycles in $DS_{0.35}$

Iteration	Total cycles	Worst arc	Appeared in (cycles)
1	3548539	AntiTheft→ RiskAversion	1376399
2	2172140	ThisCarCost→ Age	904082
3	1268058	ThisCarCost→ MakeModel	790539
4	477519	OtherCarCost→ Age	196251
5	293739	HomeBase→ Age	156595
6	153178	OtherCarCost→ DrivingSkill	80196
7	72982	Accident→ DrivQuality	31510
8	41472	Accident→ SeniorTrain	23695
9	17777	Accident→ DrivHist	9709
10	8068	Theft→ MakeModel	4276
11	3792	RuggedAuto→ MakeModel	1903
12	1889	ThisCarDam→ DrivHist	997
13	892	DrivHist→ DrivQuality	397
14	495	SeniorTrain→ RiskAversion	215
15	280	DrivQuality→ RiskAversion	129
16	157	DrivingSkill→ RiskAversion	94
17	63	DrivingSkill→ Age	42
18	21	DrivingSkill→ DrivQuality	6
19	15	Airbag→ MakeModel	3
20	12	OtherCar→ DrivingSkill	3
21	9	Theft→ HomeBase	2
22	7	SeniorTrain→ DrivHist	2
23	5	Theft→ AntiTheft	1
24	4	Airbag→ VehicleYear	1
25	3	VehicleYear→ Antilock	1
26	2	DrivHist→ DrivingSkill	1
27	1	RuggedAuto→ ThisCarDam	1

Table B.10.: Variables removed to eliminate cycles in the $DS_{0.35}$ network structure.

C. Results of Structure Evaluation

This section contains detailed results from the evaluation in Chapter 5. Accompanying each chart are the raw numbers calculated during the evaluation.

BIC Scores of Evaluation Networks

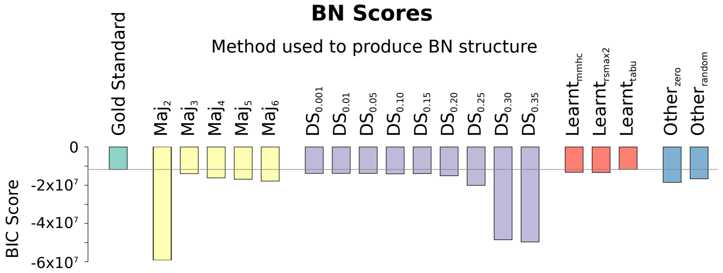


Figure C.1.: Scores calculated for the *Gold* and *Eval* networks.

(a) *Maj* structures.

Network	Score
Maj_2	-53389336
Maj_3	-13962081
Maj_4	-16137169
Maj_5	-17193272
Maj_6	-17834283

(b) *DS* structures.

Network	Score
$DS_{0.001}$	-13825782
$DS_{0.01}$	-13813215
$DS_{0.05}$	-13814269
$DS_{0.10}$	-14057724
$DS_{0.15}$	-13900837
$DS_{0.20}$	-15051534
$DS_{0.25}$	-20054815
$DS_{0.30}$	-48522871
$DS_{0.35}$	-49668297

(c) Other structures.

Network	Score
Gold	-11721585
$Learnt_{mmhc}$	-13051390
$Learnt_{rsmax2}$	-13137048
$Learnt_{tabu}$	-11665848
$Other_{Rand}$	$\mu = -17469564$
$Other_{Zero}$	-18516808

Table C.1.: BIC scores, calculated by sampling a data set from the *Gold* BN, to compare to *Eval* BNs. See Figure 5.15 (p132) for details on parameterisation.

Number of Arcs in Evaluation Networks

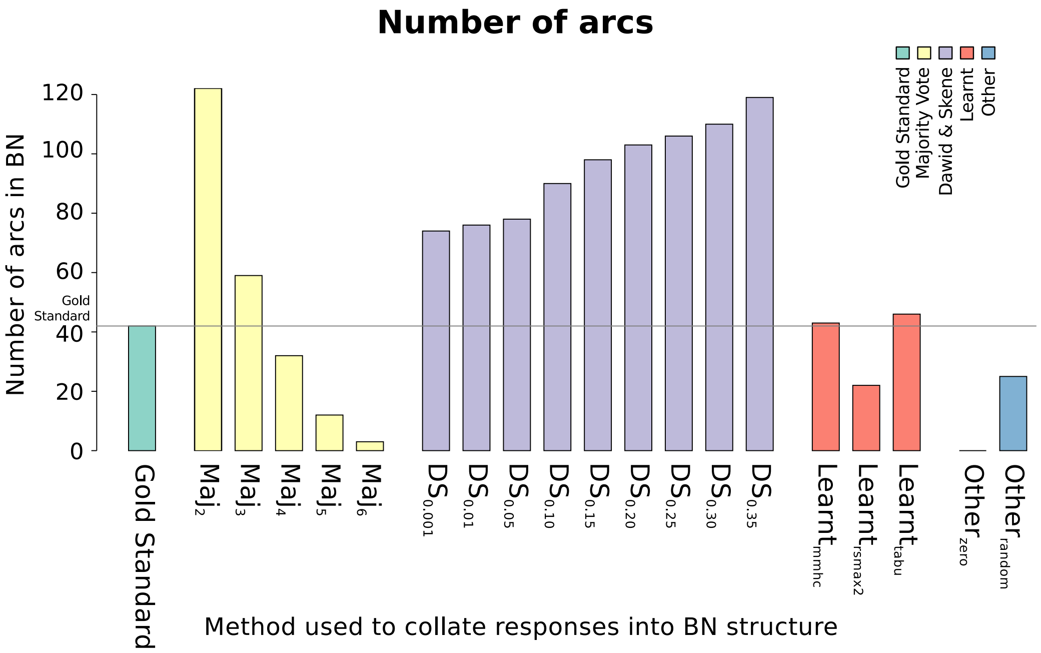


Figure C.2.: Number of arcs in the *Gold* and *Eval* networks.

(a) *Maj* structures.

Network	Score
Maj_2	121
Maj_3	58
Maj_4	31
Maj_5	11
Maj_6	3

(b) *DS* structures.

Network	Score
$DS_{0.0001}$	0
$DS_{0.001}$	74
$DS_{0.01}$	76
$DS_{0.05}$	78
$DS_{0.10}$	90
$DS_{0.15}$	98
$DS_{0.20}$	103
$DS_{0.25}$	106
$DS_{0.30}$	110
$DS_{0.35}$	119

(c) Other structures.

Network	Score
Gold	52
$Learnt_{mmhc}$	46
$Learnt_{rsmax2}$	20
$Learnt_{tabu}$	50
$Other_{Rand}$	$\mu = 27$
$Other_{Zero}$	0

Table C.2.: Number of arcs in the *Gold* and *Eval* networks.

SHDs of Evaluation Networks

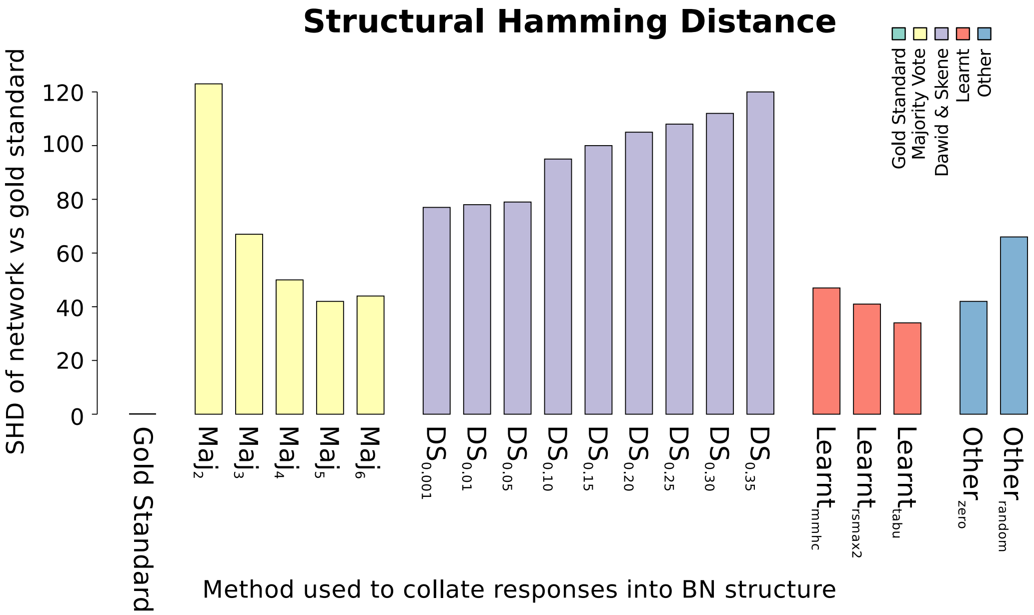


Figure C.3.: SHD of the *Eval* networks compared to the *Gold* network.

(a) *Maj* structures.

Network	Score
Maj_2	122
Maj_3	66
Maj_4	49
Maj_5	43
Maj_6	44

(b) *DS* structures.

Network	Score
$DS_{0.001}$	77
$DS_{0.01}$	78
$DS_{0.05}$	79
$DS_{0.10}$	95
$DS_{0.15}$	100
$DS_{0.20}$	105
$DS_{0.25}$	108
$DS_{0.30}$	112
$DS_{0.35}$	120

(c) Other structures.

Network	Score
Gold	0[1]
$Learnt_{mmhc}$	42
$Learnt_{rsmax2}$	40
$Learnt_{tabu}$	36
$Other_{Rand}$	$\mu = 59.1$
$Other_{Zero}$	42

Table C.3.: SHD of the *Eval* networks compared to the *Gold* network.

F1 Scores of Evaluation Networks

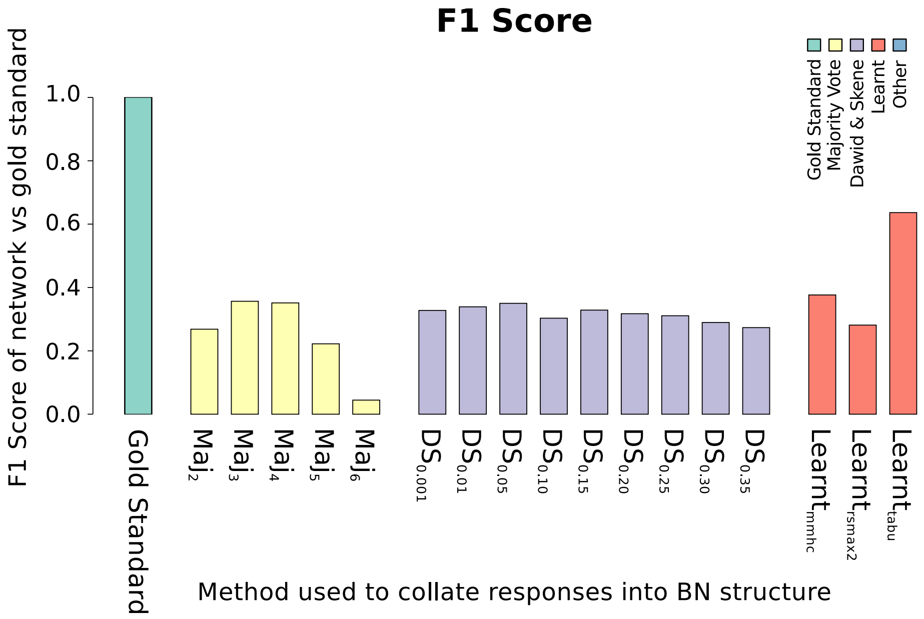


Figure C.4.: F1 score for the *Eval* networks when compared to the *Gold* network.

(a) *Maj* structures.

Network	Score
Maj_2	0.268
Maj_3	0.356
Maj_4	0.351
Maj_5	0.222
Maj_6	0.044

(b) *DS* structures.

Network	Score
$DS_{0.001}$	0.328
$DS_{0.01}$	0.339
$DS_{0.05}$	0.350
$DS_{0.10}$	0.303
$DS_{0.15}$	0.329
$DS_{0.20}$	0.317
$DS_{0.25}$	0.311
$DS_{0.30}$	0.289
$DS_{0.35}$	0.273

(c) Other structures.

Network	Score
Gold	1
$Learnt_{mmhc}$	0.376
$Learnt_{rsmax2}$	0.281
$Learnt_{tabu}$	0.636

Table C.4.: F1 score for the *Eval* networks when compared to the *Gold* network.

D. Issues Surrounding Choice of Gold Standard BN

Section 5.1.3 (p105) discussed in detail the choice of the car insurance network from Binder et al. (1997) as the gold standard. This section discusses some problems that arose due to the way in which the evaluations in this thesis changed the gold standard, and also issues intrinsic to the network itself.

D.1. Differences Between Gold Standard BNand Evaluation Network

At the outset of the evaluations, a decision was made to remove some information from the gold standard network, to make it more appropriate for usage with a cohort of participants from Victoria, Australia. The following sections discuss these decisions in the context of the BN structure elicitation survey presented in Chapter 5 and CPT elicitation survey in Chapter 7 respectively.

D.1.1. Variables Removed for Structure Elicitation

Of the original 27 variables in the car insurance network, two were removed. These were deemed to be unneeded in the context of an Australian study.

Medical Cost

Although insurers are indeed interested in medical costs in Victoria, it is likely not as important as in the United States. This is due to the presence of the Traffic Accident Commission (TAC) in Section 5.1.3 (p105) discussed in detail the choice of the car insurance network (Binder et al., 1997) as the gold standard. and comparable organisations in other states, which pay for injuries arising from traffic accidents. When registering a vehicle, a mandatory premium is paid to the TAC. Due to this, the medical cost need not always be paid by private insurance companies. Thus, it seemed counter productive to include the *Medical Cost* variable in a survey about insurance risk assessments.

Cushioning

By inspecting the structure of the car insurance network, it seems likely that the decision to include the *Cushioning* variable in the original BN was motivated by its ability to help when reasoning about the *Medical Cost* variable. Once the *Medical Cost* variable was removed, the *Cushioning* variable did little to impact the remaining network.

D.1.2. Variable States Removed for CPT Elicitation

For the CPT evaluation study, 3 out of 88 variable states were removed, in addition to the two variables discussed above. These were removed either to bring the variables more in line with an Australian cohort of participants, or to reduce the potential for generating confusing questions.

Make/Model

$\{Sports\,Car, Economy, Family\,Sedan, Luxury, Super\,Luxury\}$

The *Super Luxury* state was removed as the similarity of *Luxury* and *Super Luxury* was not great enough to warrant a further state.

Car Value

$\{Five Thousand, Ten Thousand, Twenty Thousand, Fifty Thouand, Million\}$

Almost all other variables relating to car cost had the following states:

$$\{Thousand, Ten Thousand, Hundred Thousand, Million\}$$

The *Car Value* variable was brought in line with these, by opting for *Thousand* instead of *Five Thousand*. Likewise, *Twenty Thousand* and *Fifty Thousand* were replaced with *Hundred Thousand*. In hindsight, this was a mistake as the value of a car will likely be less than other values considered by an insurance company (e.g. total cost to insurer), and should have remained discretized into smaller groups as it was originally.

Socio Economic Status

$\{Prole, Middle, Upper Middle, Wealthy\}$

The *UpperMiddle* state was removed as it was deemed to be something which was not well enough defined to ask the participants about. Even with this state removed, some participants still commented that "The initial section categorises people whcih [sic] i [sic] did not like to do". This type of concern could be alleviated in future studies by more training of participants, thus being able to calibrate their expectations of the type of questions that require answering and why.

D.1.3. Issues Arising due to Variable and State Removal

At the time the decision was made to remove the small number of variables and states above, it was deemed that the removal would not have an effect on the evaluation. The reasoning was that because variables were being removed and not added, they could also be removed form any test data sets used to evaluate the resulting survey BN. However, perhaps counter-intuitively, removing only one state causes more trouble for evaluation than does removing an entire variable. For example, consider the following data sets shown in Table D.1. The original

Table D.1.: Ramifications of removing a single variable vs removing a single state.

(a) Sampled data from the *Gold* BN.

Socio Economic Status	Driving Skill	Car Value	Cushioning	Airbag
Middle	Normal	$1,000	Poor	False
Upper Middle	Normal	$10,000	Fair	True
Wealthy	Normal	$100,000	Excellent	True

(b) Removing an entire variable from the data set.

Socio Economic Status	Driving Skill	Car Value	~~Cushioning~~	Airbag
Middle	Normal	$1,000	~~N/A~~	False
Upper Middle	Normal	$10,000	~~N/A~~	True
Wealthy	Normal	$100,000	~~N/A~~	True

(c) Removing an individual state from the data set.

Socio Economic Status	Driving Skill	Car Value	Cushioning	Airbag
Middle	Normal	$1,000	Poor	False
~~*N/A*~~	Normal	$10,000	Fair	True
Wealthy	Normal	$100,000	Excellent	True

data set in Table D.1a has all of the variables and states that exist in the *Gold* BN. When removing the *Cusioning* variable in Table D.1b, the data set is still a complete data set, albeit with one less column. However, when removing only an individual state (Table D.1c), the resulting data set contains missing values, as the entire column wasn't removed.

Most software for comparing a BN to a data set in order to see how well the BN models the data, require the BN and the data set to have the same set of variables and states. Ensuring the data sampled from the *Gold* network corresponds to the *Eval* network structure requires reconciling the same variables. Given the evaluation contained two less variables than the *Gold* network, this involved removing the two variables from any sampled data (e.g. Table D.1b). The absence of an entire column from a data set does not prove to be problematic.

Removing individual states, however, proves more troublesome. Each data point that corresponds to a removed state must also be removed (Table D.1c). The question then becomes, what is it replaced with, or is it replaced at all? Should the resulting row be discarded entirely? Would discarding the entire row cause a bias against samples which tend to gravitate toward the removed states? These questions are all the exact same questions asked of statisticians when dealing with missing data. The only difference is that the missing data was induced due to an error on behalf of the researcher in this project, rather than some stochastic process or measurement error.

Solving the Missing Data Problem

It was decided that the rows with missing data should remain. To illustrate, consider the removal of the *Upper Middle* state of *Socio Economic Status*. If each entire row containing the *Upper Middle* state was removed, it would likely have inadvertently created a bias against other variable states expected of wealthy car owners, such as *Luxury Make/Models*.

To retain rows with missing data, the data must be replaced in a principled manner. One suitable method for replacing such data is Multiple Imputation (MI), discussed in detail by Schafer and Graham (2002) and others. MI involves each missing piece of data being imputed several times using a Bayesian approach. Any analysis that was planned for the entire data set can then be conducted on the imputed data sets. This is done several times, for different imputations, before the results are then aggregated and averaged.

D.2. CPTs Which Were Evaluated

Due to the removal of variables discussed above, the evaluation of CPTs elicited using SEBN (Chapter 7) was only able to be applied to 11 out of the 27 variables in the gold standard BN. Table D.2 shows the list of variables which were analysed, and lists why the excluded variables were not analysed.

D.3. Other Issues with Gold Standard BN

In addition to the changes to the network discussed above, there are also some other problems with the car insurance network Binder et al. (1997). In hindsight, these issues indicate that it was perhaps not the optimal choice for a gold standard.

Exhaustive Variable States

For variables to be useful in BNs, the states they can take should be exhaustive and mutually exclusive. However, some of the states in the car insurance network are not exhaustive, for example:

Make/Model $\{Sports\,Car, Economy, Family\,Sedan, Luxury, Super\ Luxury\}$

Car Value $\{Five\,Thousand, Ten\,Thousand, Twenty\,Thousand, Fifty\,Thouand, Million\}$

There are clearly more car types than just those specified by *Make/Model*. Perhaps a bit more subtly, the Car Value could either be taken to represent $< Five\,Thousand$ up to $< Million$ (which excludes values above one million), or $> Five\,Thousand$ up to $> Million$ (which excludes values less than five thousand).

Mutually Exclusive Variable States

In addition to not being exhaustive, some variables do not encode mutually exclusive states, such as:

Home Base $\{Secure, City, Suburb, Rural\}$

The states of the Home Base variable are not mutually exclusive, as it is very likely that there is secure parking in the city, in the suburbs, or in rural areas.

Age of the Network

The car insurance network was published in 1997, and thus only has a small number of safety features incorporated into the network (e.g. *Airbag*). Since 1997, several new safety features have either been introduced, or made their way into a greater number of consumer automobiles. These features include Electronic Stability Control, Antilock Brakes, Adaptive Cruise Control, and others. This may have changed the way in which participants answered questions during SEBN about the chance of an accident occurring.

(a) Variables with no parents.

No parents
Age Mileage

(b) Variables with a single parent.

Single parents	Excluded
ILiCost	
SocioEcon OtherCar	Related to *SocioEcon*, which had a different CPT between *Gold* and *Survey*

(c) Variables with multiple parents.

Multiple parents	Excluded
DrivingSkill Accident SeniorTrain DrivQuality DrivHist ThisCarDam OtherCarCost PropCost	
GoodStudent RiskAversion AntiTheft HomeBase VehicleYear MakeModel	Related to *SocioEcon*, which had a different CPT between *Gold* and *Survey*
MedCost Cushioning	MedCost irrelevant to Australia and Cushioning only impacts MedCost
RuggedAuto Antilock Airbag Theft ThisCarCost	Other misc data collection errors in BNE software caused some elicited values to not be submitted.

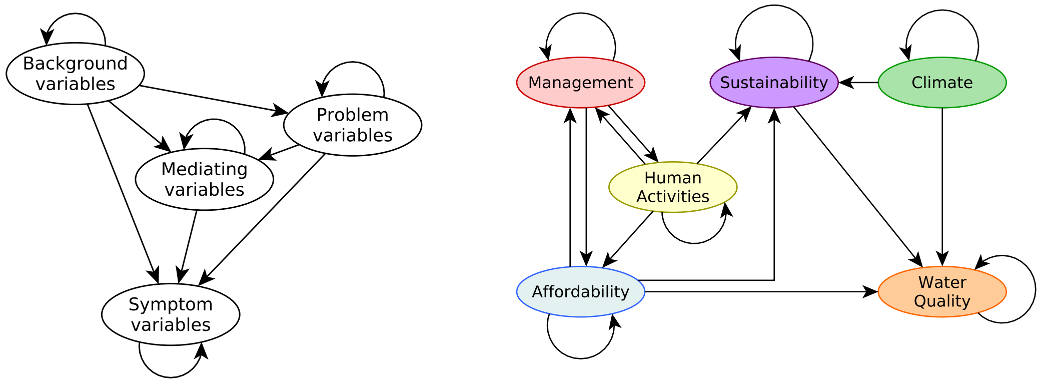
Table D.2.: The variables which were used in the evaluation in Chapter 7.

E. Alternative Variable Classes

SEBN discussed in Chapter 4 discusses a technique proposed by Kjærulff and Madsen (2008) for constraining the possible questions when eliciting a BN structure. This appendix performs a post-hoc analysis of the water management BN, developed by Chan et al. (2010) using traditional KEBN. The goal was to see how the magnitude of the SEBN task could be reduced if the survey questions were based on domain specific variable classes.

The entire BN from Chan et al. (2010) is shown in Figure E.2, including all 49 variables. If the naive adjacency-matrix question generation approach was used (e.g. Xiao-xuan et al., 2007), then all 49 variables would be eligible to depend on all other variables, resulting in 2352 questions ($n^2 - n$).

Figure E.1.: Comparison between generic and domain specific variable classes, and the inter-class dependencies they exhibit.



(a) Generic variable classification scheme, adapted from from Kjærulff and Madsen (2008) and discussed in Section 4.3 (p80).

(b) Domain specific classes identified in the Chan et al. (2010) BN.

Class	Variables in class	Variables that can influence those in class	Resulting survey questions
Management	10	28	270
Sustainability	14	35	476
Climate	3	3	6
Human Activities	8	18	136
Affordability	10	18	170
Water Quality	4	31	120
Total	49	N/A	1178

Table E.1.: Number of variables in each domain specific class, and the resulting number of survey questions required.

However, Chan et al. (2010) discussed some key classes of variables that appear in the network (Figure E.2b). The inter-class dependencies were ascertained during this analysis by analysing the dependencies between each variable class in the final BN. The way in which each variable was classified for the purpose of this analysis is shown in Figure E.2.

When constraining the generated SEBN questions based on these domain specific classes, the number of possible relationships is constrained by almost exactly 50% (from 2352 possible questions to 1178, Table E.1).

Note on Domain Specific Variable Classification

The decision as to whether a variable class from Chan et al. (2010) can influence others was deduced by looking at the BN in Figure E.2. Any variables which influenced those in a separate class induced a dependency between those two classes. Note that in a real SEBN project, this is not how class dependencies are inferred, because a causal BN likely doesn't yet exist. Rather, experts would be employed to discuss the potential for causal dependencies between variables of differing classes. Thus, this analysis and the subsequent reduction in survey questions should be treated as post-hoc analysis. Further research should be done to investigate the use of domain specific variables, and how successful they are (or are not) at reducing the amount of questions required, or allowing experts to be allocated questions in their specific area of expertise.

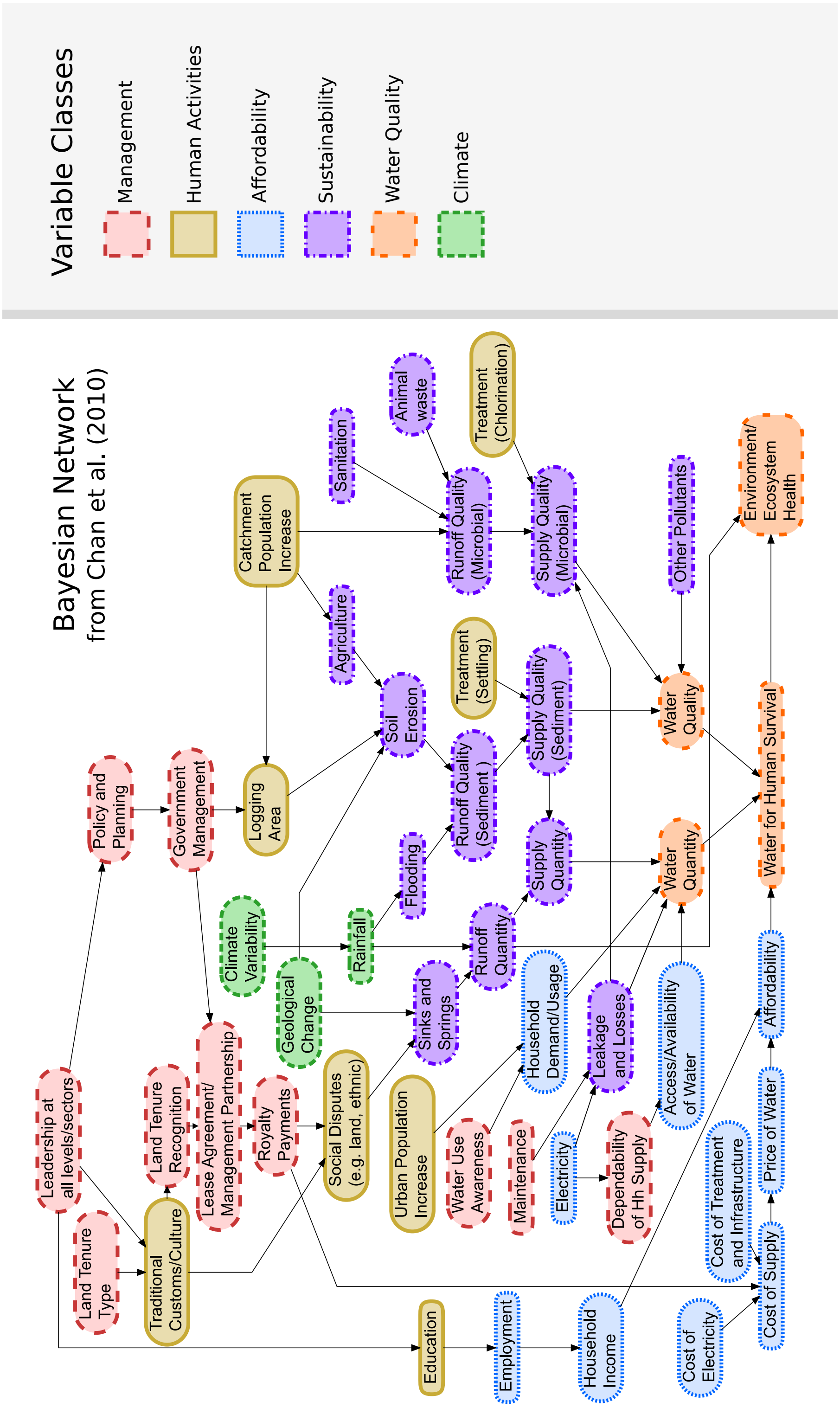


Figure E.2.: The network from Chan et al. (2010) after manually classifying variables into six different categories for the purpose of this analysis.

F. Question Generation Examples

Shown in this appendix is a collection of survey questions that were asked as part of the evaluations in Chapter 5 and 7. The questions were generated using SEBN, based on the variables and networks structure of the car insurance network (Binder et al., 1997).

F.1. Structure Elicitation Questions

Section 4.4.1 (p86) discussed how the generated questions in BNE are grouped together based on variables in an adjacency matrix. For each column in the matrix, a question such as : "Do any of the following $\{X, Y, Z\}$ influence A?" is generated, where A is the column of the matrix, and $\{X, Y, Z\}$ are the rows (excluding A). Listed below is a selection of questions that resulted from the car insurance network using this question format. Each potential child variable was listed below the question, with an option for "Yes" or "No" (Figure F.1).

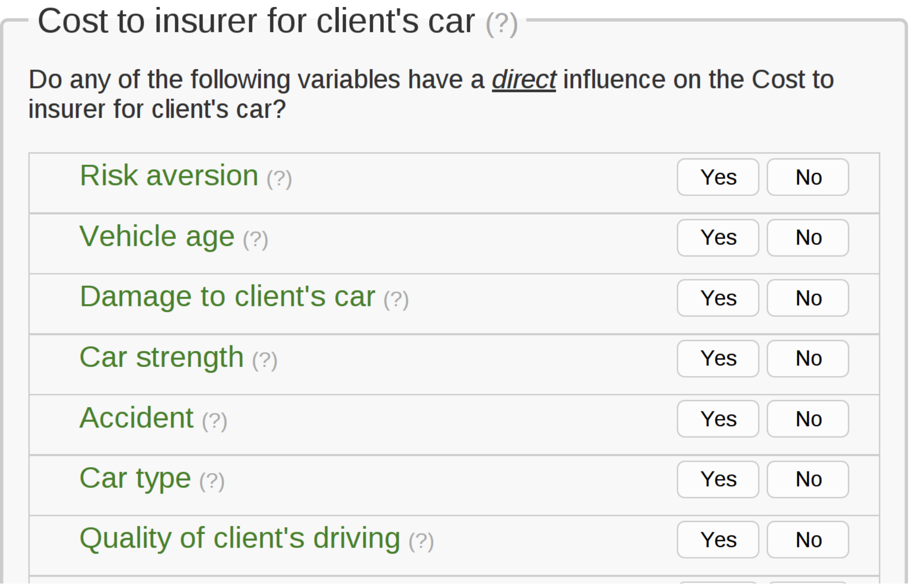
GoodStudent "Do any of the following have a *direct* influence on the fact that the client was a good student when the learnt to drive ?"

Age "Do any of the following have a *direct* influence on the Age of the client?"

RiskAversion "Do any of the following have a *direct* influence on the Risk aversion of the client?"

VehicleYear "Do any of the following have a *direct* influence on the age of the client's vehicle?"

ThisCarDam "Do any of the following have a *direct* influence on whether a client's car is damaged?"



Cost to insurer for client's car (?)

Do any of the following variables have a *direct* influence on the Cost to insurer for client's car?

- Risk aversion (?)
- Vehicle age (?)
- Damage to client's car (?)
- Car strength (?)
- Accident (?)
- Car type (?)
- Quality of client's driving (?)

Figure F.1.: Screenshot from BNE software configured with variables from the insurance network.

RuggedAuto "Do any of the following have a *direct* influence on the strength of the client's car?"

Accident "Do any of the following have a *direct* influence on whether the client becomes involved in an accident?"

After using SEBN to elicit the structure of a BN in Chapter 5, it became evident that the phrasing of the questions appeared backward. Instead of asking: "Do any of the variables below this have an influence on this", they should have asked "Does this variable influence any of those below". The BNE software has been changed to reflect this for future elicitations.

F.2. Probability Elicitation Questions

When generating questions for CPT elicitation using SEBN, Section 6.2 (p156) discusses how the choice of questions depends on how many parents a variable has in the BN structure.

F.2.1. Variables with Zero Parents

In the case of variables without any parents in the BN structure, the marginal probability of each state is explicitly elicited (Section 6.2.1, p157). The car insurance network had two variables without parents, and below are the questions required to elicit the probabilities of each of their states.

Age **Pr(Age=Adolescent)** “What is the likelihood of the following scenario: Client is a *young adult*?"

- **Pr(Age=Adult)** "What is the likelihood of the following scenario: Client is an *adult?”*

- **Pr(Age=Senior)** “What is the likelihood of the following scenario: Client is a *senior*?"

Mileage **Pr(Mileage=FiveThou)** "What is the likelihood of the following scenario: Client’s car has driven *less than 10,000km*?"

- **Pr(Mileage=TwentyThou)** "What is the likelihood of the following scenario: Client’s car has driven *between 10,000km and 20,000km*?"

- **Pr(Mileage=FiftyThou)** "What is the likelihood of the following scenario: Client’s car has driven *between 20,000km and 50,000km*

- **Pr(Mileage=Domino)** "What is the likelihood of the following scenario: Client’s car has driven *over 50,000km*?”

F.2.2. Variables with One Parent

Variables with a single parent had their full CPT elicited explicitly, as described in Section 6.2.2 (p160). One such variable was ILiCost (“Insurance Liability Cost”). Below is a sampling of some questions required to elicit $\Pr(ILiCost|Accident)$ using SEBN.

ILiCost **Pr(ILiCost=Thousand|Accident=Zero)**

- What is the likelihood of the following scenario?
  - Client will *claim less than $1,000* to fix *buildings or property* damaged in an accident.

- If we know that:
  - Client will *not get into an accident*

Pr(ILiCost=Thousand|Accident=Mild)

- What is the likelihood of the following scenario?
  - Client will *claim less than $1,000* to fix *buildings or property* damaged in an accident.
- If we know that:
  - Client will get themselves into a *mild accident*

Pr(ILiCost=TenThou|Accident=Zero)

- What is the likelihood of the following scenario?
  - Client will claim *between $1,000 and $10,000* to fix *buildings or property* damaged in an accident
- If we know that:
  - Client will *not get into an accident*

F.2.3. Variables with Multiple Parents

Finally, variables with multiple parents follow the weighted sum algorithm in conjunction with AHP (Section 6.2.3, p162). This consists of three different phases, the first is to elicit the CPCs for each state of each parent. The second stage is to elicit the conditional probabilities of each child state, conditioned on each of those CPCs. Finally, the weight of each variable is calculated by asking for pairwise comparisons between each of the parent variables. For reference, questions regarding the CPT of the $CarValue$ variable will be used in this section (Figure F.2).

Compatible Parent Configurations

Below is the question required to elicit the most compatible two states for the *VehicleYear* and *MakeModel* variables, when *Mileage* is *TenThou*. This needs to be repeated for each state of *Mileage*, and then for each state of the *VehicleYear* and *MakeModel* variables.

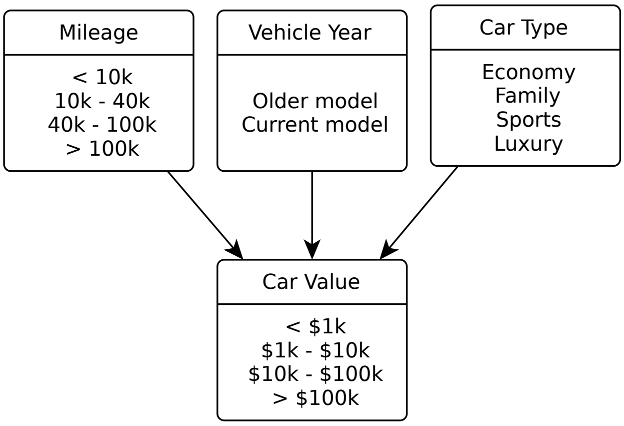


Figure F.2.: The $CarValue$ variable has three parents, which means the CPT is elicited using the weighted sum algorithm when using SEBN.

Mileage= CPC(Mileage=TenThou)

- If Client's car has driven *less than 10,000km*, then I expect:
  - Client's car is a *current* model
  - Client's car is an *older* model
- And
  - Client's car is a *sports car*
  - Client's car is an *economy car*
  - Client's car is a *family sedan*
  - Client's car is a *luxury car*

Child Probabilities, Conditioned on CPCs

Once the relevant CPCs have been elicited, then the conditional probability of the child states can be elicited. The following assumes that the probability being elicited is the $\Pr(CarValue = Thou|CPC(Mileage = TenThou))$.This also presumes that the CPC of $Mileage = TenThou$ was elicited as $VehicleYear = Current$ and $MakeModel = FamilySedan$ by answering the question above:

- What is the likelihood of the following scenario?

  - "Clients car is worth less than $1,000 (at time of insuring)"

- If we know that:
  - Client's car has driven *less than 10,000km*
  - Client's car is a *current* model
  - Client's car is a *family sedan*

Pairwise Comparisons

To complete the weighted sum algorithm for the $CarValue$, the following three questions are required in order to perform AHP and obtain relative weights of each parent:

Mileage vs VehicleYear

- Which influences the *(Monetary) value of client's car* more?
  - Vehicle Age
  - Mileage
  - Both have the same influence

Mileage vs MakeModel

- Which influences the *(Monetary) value of client's car* more?
  - Mileage
  - Car type
  - Both have the same influence

VehicleYear vs MakeModel

- Which influences the *(Monetary) value of client's car* more?
  - Vehicle Age
  - Car type
  - Both have the same influence