MINING SURPRISING PATTERNS

by

Yen-Ting Kuo

Submitted in Total Fulfilment of the Requirements for the Degree of Doctor of Philosophy
November 2009

Department of Information Systems
University of Melbourne

(Produced on archival quality paper)
Abstract

From the perspective of an end-user, patterns derived during the data mining process are not always interesting. The mining of unexpected patterns is a computational technique introduced in earlier work to address this problem. However, such unexpected patterns are not necessarily surprising to the user.

In this thesis, we show that the quality of a user’s knowledge, that is encoded in computational form, is key to bridging the gap between unexpected and surprising patterns. The thesis presents an approach that reduces this gap by exploiting a synergy between existing techniques utilised in data mining. Key to the new approach is (1) the employment of a domain ontology to guide the mining of association rules, (2) an encoding of users’ knowledge using a Bayesian network representation, and (3) a probabilistic model to generate explanations for unexpected rules.

The methods are tested on real-world data in two domains using users who are domain experts. In the medical domain, a dataset of chronic kidney disease patients is mined with a nephrologist; in the educational domain, a dataset of a decimal comparison test of children is mined with two education researchers. Surprising patterns have been successfully discovered. Further gaps, identified during the investigation, are captured in the discussion of case studies. In total, the surprisingness problem needs to be tackled from the aspects of knowledge representation, knowledge acquisition, interpretation assistance; and prevention of meaningless rules. A lack of sufficient information about rules is found to be a major cause of meaningless rules, where the uninformativeness problem was caused outside the scope of rule ranking. We conclude that the surprisingness problem should be further researched beyond the scope of the thesis.
Declaration

This is to certify that

- the thesis comprises only my original work towards the PhD except where indicated in the Preface,

- due acknowledgement has been made in the text to all other material used,

- the thesis is less than 100,000 words in length, exclusive of tables, maps, bibliographies and appendices.

Yen-Ting Kuo
Preface

During the course of this research, a number of public presentations have been made which are based on the work presented in this thesis. They are listed here for reference.

  ........................................................................................................... Partially incorporated in Ch. 6.

  ........................................................................................................... Partially incorporated in Ch. 4.
Acknowledgments

First, my heartfelt thanks to my principal supervisor, Professor Liz Sonenberg, for her generous offer of the PhD position so that I can fulfil my dream of postgraduate study. To my supervisors, Liz Sonenberg, Andrew Lonie and Adrian Pearce, thank you for your support and guidance throughout this research: for constantly guiding me toward exploration in the right directions; for questioning me about the unclear key thoughts; and for shaping my ambiguous concepts by interpreting the research from different perspectives. Thank you also for trusting me to work in my own time, in my own way, and for always encouraging me and building up my confidence. Lastly, thank you for your help, especially Andrew, in correcting and editing the manuscript of the thesis; it is very important to me whose native language is not English.

To my Advisory Committee Member, James Bailey, thank you for providing the clear head needed to help keep the overall story in focus. It seems like I entered each of our progress meetings a little unsure about where the scope of research should be, but left each with a renewed focus and a clear picture of the road ahead; thank you for the codes of emerging pattern mining which contributes to my preliminary analysis of data.

To the experts who participated in the case studies, Kathy Paizis, Kaye Stacey and Vicki Steinle, thank you for spending lots of time in discussing the DM results and in elaborating your knowledge to me. From every discussion we had, your comments sparked new ideas and deeper understanding which inspired the technique and shaped the findings of the thesis. Many thanks to Ann Nicholson, for your kindly discussion and suggestions about my work; shaping my understanding in Bayesian networks; for providing your insights into the manuscript and case study.

I must also acknowledge the generous support of the University of Melbourne in providing the Melbourne International Research Scholarship (MIRS) and the Melbourne International Fee Remission Scholarship (MIFRS), without which this research would not have been possible.
To my fellow colleagues and friends, Samin, Budhi, Daghan, Raymond, Zhiqi, Bin, Elsa, Jian, and Jens, thanks for making the whole postgraduate experience so rewarding, and for your kindly help whenever it was needed. To Elsa, thank you for your help in the set up of the codes of emerging pattern mining, and the discussion of data mining ideas. To Jian, thank you for the discussion on reasoning in logic and inductive logic programming. Many thanks to Zhiqi for being such a good friend to me; it is very joyful to share our views of being PhD students and of life in Melbourne. Special thanks to my buddy Yuan-Pu for your encouragements and heartwarming greetings from the UK. My thanks also to the staff of the Department of Information Systems: James, Rhonda, Aaron, Emin, Mathew, Marg, Jenny, Liam and the others, for your professional assistance throughout my studies in the department.

To my parents, my elder sister, parents in law, and sisters/brothers in law, thank you for your unwavering support – financial, physical and emotional – throughout all of my studies. Your blessing gives me the energy for the pursuit of my PhD.

Finally, and of course most importantly, to my wife Yia-Fang – thank you for your unending love and understanding; for your company with me all the time; and for the delivery of our little angel – our son Zeta. Thank you for giving me a complete family throughout my studies in Melbourne.
Contents

Abstract iii
Declaration iv
Preface v
Acknowledgments vii
List of Figures xv
List of Tables xvii

1 Introduction 1
1.1 Problem Statement 4
1.2 Scope 5
1.2.1 Domain Knowledge 6
1.2.2 Unexpectedness 6
1.2.3 Surprise 7
1.2.4 Knowledge Acquisition 7
1.3 Research Question 8
1.4 Thesis Contribution 9
1.5 Thesis Outline 11

2 Knowledge Rich Data Mining 13
2.1 Overview 13
2.2 Subjective Interestingness 17
2.2.1 Justification of Terminology 18
2.3 Data-Driven Surprisingness 21
2.3.1 Exception Rules 21
2.3.2 Simpson’s Paradox 24
2.4 Review of Unexpectedness 25
## CONTENTS

2.4.1 Knowledge Representation Techniques ................................. 26  
2.4.2 Definitions and Calculations of Unexpectedness ......................... 32  
2.4.3 Knowledge Acquisition and Update .................................. 36  
2.4.4 More Related Work ............................................. 42  
2.4.5 Comparison of Knowledge Representations ............................ 42  
2.4.6 Principles for Unexpectedness .................................... 48  
2.4.7 Summary ...................................................... 49  
2.5 Explanations for Association Rules .................................... 50  
2.5.1 Summary ...................................................... 52  
2.6 Knowledge Acquisition ................................................ 52  
2.6.1 Knowledge Acquisition Bottleneck .................................. 55  
2.6.2 Summary ...................................................... 58  
2.7 Ontology in Data Mining ............................................... 59  
2.7.1 Domain Ontology ................................................ 59  
2.7.2 Data Mining Ontology ............................................ 61  
2.7.3 Summary ...................................................... 61  
2.8 Chapter Summary .................................................... 62  

### 3 Datasets

3.1 Materials ............................................................. 63  
3.1.1 ANZDATA dataset ............................................... 65  
3.1.2 DCT dataset ................................................... 66  
3.1.3 KSL dataset .................................................... 69  
3.1.4 Census Income dataset ......................................... 71  
3.1.5 WWW User Survey dataset .................................... 71  
3.2 Methods ............................................................. 71  
3.3 Chapter Summary .................................................... 73  

### 4 Ontology-Driven Data Mining

4.1 Design I - Ontology for Semantic-Rich Association Rules ............... 77  
4.1.1 Overall Procedure ............................................... 78  
4.1.2 Case Study: Nephrology ........................................... 79  
4.2 Design II - Ontology for Unexpected Rules ................................ 87  
4.3 Discussion and Conclusion ............................................. 88
## 5 Unexpected Pattern Mining 90

5.1 Motivation ................................................................. 90
5.2 Problem Statement ...................................................... 90
5.3 Proposed Design ......................................................... 91
  5.3.1 Algorithms for Unexpectedness .............................. 91
  5.3.2 Association Rules and Correlations ....................... 94
  5.3.3 Acquiring User’s Knowledge ................................. 95
  5.3.4 Local Bayesian Networks for Rules ....................... 97
  5.3.5 Computability of Unknown ................................. 99
  5.3.6 Overall Design of Unexpected Rule Mining .............. 100
5.4 Case Study ............................................................... 101
  5.4.1 Danish Geriatric (KSL) ....................................... 101
5.5 Chapter Summary ....................................................... 106

## 6 Explanation Generation 108

6.1 Motivation ............................................................... 109
6.2 Definition ................................................................. 109
  6.2.1 Modeling Explanations in Bayesian Networks .......... 112
  6.2.2 Graphical Semantics of Bayesian Networks .......... 114
6.3 Explanation-Generating Algorithm ............................... 117
  6.3.1 Discussion ......................................................... 118
  6.3.2 Ranking Function .............................................. 122
6.4 Related Work ............................................................ 124
  6.4.1 Causal Diagrams in Epidemiology ....................... 124
  6.4.2 Structure Learning of Bayesian Networks ............. 126
6.5 Case Studies ............................................................. 127
  6.5.1 Case Study I: WWW User Survey ......................... 128
  6.5.2 Case Study II: Census Income ............................. 130
6.6 Chapter Summary ........................................................ 137

## 7 Case Studies 139

7.1 Case Study I: Nephrology ........................................... 139
7.2 Case Study II: Education in Mathematics .................... 146
  7.2.1 The Medium Level of Types 1 and 2 ................. 151
  7.2.2 Analysis of Rule \{Q_{18r_2} = 0, Type_5 = 2\} .......... 155
7.3 Evaluation of Experiments .......................................... 158
List of Figures

1.1 Relation map of interestingness and surprisingness in the aspects of users and KDD systems. ........................................ 3
1.2 Scope of thesis in KDD. ................................................... 5
2.1 Comparison between objective and subjective interestingness. . 18
2.2 The nomenclature of interestingness used in the thesis. ........ 19
2.3 The categories of exception rules (Suzuki and Zytkow, 2005). . 23
2.4 Knowledge dynamics in unexpected pattern mining. ............. 26
2.5 Example of Bayesian networks. ........................................ 31
2.6 BN representing the DK of the loan-screening dataset example. 44
2.7 Transitivity property of rules. ......................................... 47
2.8 Explanation-oriented data mining framework (Yao and Zhao, 2005). ................................................................. 51
2.9 Explanation templates of (Svátek et al., 2005a). ................. 52
2.10 The categories of knowledge in engineering design (Ishino and Jin, 2002). .......................................................... 54
2.11 The integrated knowledge acquisition system in Xing et al. (2003). 56
4.1 Diagram of the mapping from attributes to classes of an ontology. 78
4.2 Portion of the Semantic Network in UMLS. ....................... 80
4.3 Result of mining risk factors of cardiovascular death ............. 85
5.1 Run time test of Algorithm 1 and 4 in mining the ANZDATA. . 94
5.2 The implication of independent nodes. .............................. 100
5.3 Procedural diagram of the design for unexpected pattern mining. 101
5.4 The initial Bayesian network of KSL. ............................... 103
5.5 First modification of BN by our approach. ......................... 104
5.6 First modification of BN by Jaroszewicz and Simovici (2004). . 104
5.7 Second modification of BN by our approach. ...................... 105
5.8 Second modification of BN by Jaroszewicz and Simovici (2004). 105
5.9 Earlier discovery of unexpected rules of KSL. .................................. 107

6.1 Inferred conditional probabilities of different intermediates. . . . . 111
6.2 Inferred conditional probabilities of different intermediates. . . . . 112
6.3 Imaginary example of complex explanation. ................................. 113
6.4 Examples of the skeletons of Bayesian networks as explanations. 115
6.5 Example of causal diagram. ..................................................... 125
6.6 Integrated unexpectedness design. ......................................... 128
6.7 Explanation of rule 1 in WWW User Survey. ................................. 129
6.8 Explanation of rule 2 in WWW User Survey. ................................. 130
6.9 Initial BN of Census Income. ................................................. 131
6.10 Explanations of rule 1 in Census Income. ................................. 132
6.11 Explanations of rule 2 in Census Income. ................................. 133
6.12 The updated Bayesian network in first iteration. .......................... 134
6.13 Updated BN for comparison in first iteration. .............................. 134
6.14 Explanations of rule 4 in Census Income. ................................. 135
6.15 Explanations of rule 6 in Census Income. ................................. 136
6.16 The updated Bayesian network in second iteration. ....................... 137
6.17 Updated BN for comparison in second iteration. ............................ 137

7.1 Local BN for Rule 5. .............................................................. 143
7.2 Local BN for Rule 6. .............................................................. 143
7.3 Explanation for Rule 5 of ANZDATA. ....................................... 144
7.4 Explanation for Rule 6 of ANZDATA. ....................................... 145
7.5 The Bayesian network serving as background knowledge. ............... 147
7.6 Explanation for rule 1 in DCT. ................................................. 148
7.7 Explanation for rule 2 in DCT. ................................................. 148
7.8 Explanation for rule 3 in DCT. ................................................. 149
7.9 Histogram of first explanation. ................................................. 150
7.10 Histogram of second explanation. .......................................... 152
7.11 The initial Bayesian network. ................................................. 152
7.12 Explanation for rule 1 in DCT. ................................................. 154
7.13 Another explanation for rule 1 in DCT. .................................... 154
7.14 Sieve diagram of Q18 and Type5. .......................................... 156
7.15 Explanation for rule 11 in DCT. .............................................. 156
7.16 Second explanation for rule 11 in DCT. .................................... 157
7.17 Sieve diagram of Q18 and Q3. .............................................. 157
### LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.18</td>
<td>Sieve diagram of $Q_3$ and $Type_5$.</td>
<td>157</td>
</tr>
<tr>
<td>7.19</td>
<td>Sieve diagram of $Q_{18}$ and $Q_3$.</td>
<td>158</td>
</tr>
<tr>
<td>7.20</td>
<td>Sieve diagram of $Type_2$ and $Type_5$.</td>
<td>158</td>
</tr>
<tr>
<td>D.1</td>
<td>A screen capture of the implemented platform.</td>
<td>200</td>
</tr>
<tr>
<td>D.2</td>
<td>An example of the local BN of a rule.</td>
<td>201</td>
</tr>
<tr>
<td>E.1</td>
<td>Example of sieve diagram.</td>
<td>203</td>
</tr>
<tr>
<td>F.1</td>
<td>A design of knowledge acquisition and unexpected rule mining</td>
<td>205</td>
</tr>
<tr>
<td></td>
<td>based on strong dependent rules.</td>
<td></td>
</tr>
<tr>
<td>G.1</td>
<td>The result of REP Tree classifies cardiovascular death in ANZ-DATA.</td>
<td>207</td>
</tr>
<tr>
<td>G.2</td>
<td>Result of temporal association rules.</td>
<td>208</td>
</tr>
</tbody>
</table>
## List of Tables

2.1 Recovery rates for the entire population. ......................... 24
2.2 Recovery rates for the sub-populations of males and females separately. ........................................ 25
2.3 Comparison of knowledge acquisition. .............................. 41
2.4 Loan-screening Dataset for comparison of unexpectedness approaches. ........................................ 43
2.5 Explanations found by the method of Yao et al. (2003). ....... 51

3.1 Reference table for datasets and their locations of case studies. 65
3.2 Variables of ANZDATA (Part I). ................................. 67
3.3 Variables of ANZDATA (Part II). ............................... 68
3.4 Fragment of the values of part of ANZDATA. .................. 69
3.5 Items of the decimal comparison test .......................... 69
3.6 Response patterns of DCT. ........................................ 70
3.7 Scores of the categorisation on item types. ..................... 70
3.8 Attributes of the KSL dataset. .................................... 70
3.9 All attributes and values of “Census Income” dataset. ......... 72
3.10 The selected questions/variables in the survey of Internet use. 73

4.1 Four-fold table for data analysis. ............................ 82
4.2 Four-fold table of a pseudo data. ................................ 83
4.3 Risk factors of cardiovascular death. ............................ 84
4.4 Comparison of risk factors of cardiovascular death. ........ 84
4.5 Some of the problems encountered during mining ANZDATA. 86

5.1 Sample of association table provided by the user. ............. 96
5.2 Comparison of unexpectedness designs. ........................ 103
5.3 Unexpected rules of first iteration. ............................. 104
5.4 Interesting itemsets of first iteration. .......................... 104
5.5 Unexpected rules of second iteration. .......................... 105
5.6 Interesting itemsets of second iteration. .......................... 105
5.7 Unexpected rules of third iteration. ................................. 106
5.8 Interesting itemsets of third iteration. ............................. 106

6.1 Interpretations of unexpected rules and explanations. ......... 121
6.2 Factors for interpretations. .......................................... 122
6.3 Two mined association rules. ......................................... 129
6.4 Explanations for rule (1) by the method of Yao et al. (2003). 130
6.5 Mined unexpected rules with $\epsilon_U = 4$ .......................... 132
6.6 Unexpected frequent itemsets in first iteration. .................. 133
6.7 Mined unexpected rules in second iteration with $\epsilon_U = 3$ .... 135
6.8 Unexpected frequent itemsets in second iteration. ............... 136

7.1 Several of the discretised attributes in the preprocessed ANZ-
DATA ................................................................. 141
7.2 Table of association graph of ANZDATA .......................... 141
7.3 Mined unexpected rules. $U$ in the last column is unexpectedness. 142
7.4 User’s feedback about unexpected rules. .......................... 143
7.5 Mined unexpected dependence rules with $\epsilon_U = 0.7$ ........... 147
7.6 Items in Types 4 and 6 ............................................. 151
7.7 Mined unexpected dependence rules that confirms the missing
link between questions and types. ..................................... 153
7.8 Unexpected dependence rules of type 2 questions. ............... 153
7.9 The contingency table of $Q19_{T_2}$ and $Q7_{T_1}$ .................. 155
7.10 Items related to the rule $\{Q18_{T_2} = 0, Type5 = 2\}$ ............ 158
7.11 The alignments between the identified gaps and the five steps
of KDD ............................................................ 164

B.1 2-way contingency table for attributes $A$ and $B$. ................ 197
Chapter 1

Introduction

Knowledge discovery and data mining (KDD), widely used in scientific discovery, business and industry domains, aims to help users gain new knowledge from data. A successful KDD trial requires a thorough understanding of the problem and thus intensive modelling and utilisation of knowledge at all stages\(^1\) (Fayyad et al., 1996). For example, data analysts (who operate the KDD system) can apply domain knowledge to the selection of relevant attributes; guiding the preprocessing and transformation steps; restricting the search space of data mining; and the user can apply domain knowledge to the interpretation of mined patterns. Domain experts (who might be the users as well) and data analysts have to share their knowledge to ensure the problem is well understood; the data is carefully selected, preprocessed and transformed; the selection of the data mining technique is appropriate; and the result is correctly interpreted. Therefore, some people argue that “data mining will always be an art” (Faloutsos and Megalooikonomou, 2007).

Towards the aim of gaining knowledge, KDD research has long devised interestingness measures for highlighting interesting patterns to users. According to Fayyad et al. (1996), interestingness is usually taken as an overall measure of pattern value, combining validity, novelty, usefulness, and simplicity.

There are two types of interestingness: objective and subjective (Geng and Hamilton, 2006; McGarry, 2005). Objective (data-driven) interestingness metrics are independent of users’ knowledge, and are defined on statistical

\(^1\)Namely, data selection, preprocessing, transformation, data mining and pattern interpretation.
significance measures such as conciseness, coverage, reliability, precision, accuracy - that is, they depend solely on the dataset being mined. *Subjective* (user-driven) interestingness metrics are dependent on the user who examines the pattern; practically, subjective measures rely on the tangible knowledge of the user where the knowledge can be formally represented in the KDD system.

*Unexpectedness*, patterns being unexpected to users, has been proposed as a major reason for why a pattern is interesting from the user’s point of view (Silberschatz and Tuzhilin, 1995, 1996). The art of mining unexpected patterns, another meaning of *unexpectedness*, computationally utilises users’ knowledge to rank the patterns, e.g. association rules. Therefore, unexpectedness is a kind of subjective interestingness and it has been researched via various approaches in last decade (Xin et al., 2006; Jaroszewicz and Scheffer, 2005; Padmanabhan and Tuzhilin, 2006, 2002, 1999, 1998; Liu and Hsu, 1996; Liu et al., 1997, 1999a, 2000; Sahar, 1999).

By earlier definition, when a pattern is *surprising* to the user, it is called *unexpected* (Silberschatz and Tuzhilin, 1995, 1996). Earlier work did not carefully build the context of these two terms; they were sometimes synonymous. For example, Carvalho et al. (2003) maintained that ‘... a rule is considered surprising to the extent that it captures knowledge that is unexpected with respect to the user’s beliefs or previous knowledge’, but other researchers have used the term *unexpected* rather than *surprising* (Geng and Hamilton, 2006).

The exact meaning of *unexpected* was ill-defined in previous research as well. Earlier work refers to ‘unexpected patterns’ as the patterns mined from a particular computational approach which can gauge the degree of unexpectedness. But, ‘unexpected’ sometimes refers to the personal judgement on rules; for example, Liu et al. (2000) stated ‘... If the unexpected rules are not truly unexpected, they serve to remind the users what they have forgotten.’ We argue that using the term *unexpected* for both computational qualities of rules and personal opinions is confusing and shall be avoided.

We reserve ‘*unexpected*’ for describing the computational feature of patterns/rules\(^2\), and ‘*surprising*’ for users’ opinions about patterns/rules. Un-

---

\(^2\)Because the thesis deals with generic rules mining in KDD, such as association rules, the term ‘pattern’ actually refers to ‘rule’ in the thesis; and we would use these two terms interchangeably.
expectedness is the computational approach for surprisingness\(^3\), but in the thesis, unexpected rules would not always be surprising to users. From the viewpoint of human judgement, a straightforward observation is that surprising is a relative term which is subject to a user’s knowledge. Therefore, the technique of mining surprising patterns would need the user’s knowledge to be encoded in order to rate the rules. The aforementioned concepts about ‘surprising’ and ‘interestingness’ are depicted in Figure 1.1 showing their relations from the aspects of users (left) and KDD systems (right). In order to articulate the concepts throughout the thesis, e.g. surprisingness, unexpectedness, user’s knowledge, encoded knowledge, the conditions for unexpectedness, we also present a nomenclature in Section 2.2.1 where the essential elements of surprisingness are discussed in detail.

There is still a big gap between subjective interestingness and users’ judgement. Narrowly speaking, this is because encoding human knowledge is limited in expressibility in any formal representation, and knowledge acquisition is al-

\(^3\)In the thesis, surprisingness refers to either (a) the task/art of mining surprising patterns or (b) patterns being surprising to users.
ways a bottleneck for constructing a high-quality knowledgebase (if we knew how to quantify its quality). Secondly, subjective interestingness also depends on the user’s specific needs (Sahar, 1999), such as the goal or interest for gaining new knowledge on a particular aspect of the data. Therefore, mining interesting patterns is never a straightforward task; it is the same for mining surprising patterns. Earlier work on unexpectedness focuses on the ranking of mined rules. However, in terms of users’ interpretation of patterns, other fundamental issues unfolded during the course of this research. The aim of the thesis is to understand the essentials for mining surprising patterns.

1.1 Problem Statement

Empirically, whether a rule is interesting or surprising to a user is not simply a yes/no question. In the author’s experience, during the development of this thesis, from working with domain experts (who are users of KDD), the domain experts have to perform complex reasoning by putting all the relevant factors and their relations together before forming a definite judgement. In the process, experts require some additional information, such as the statistical properties of the rule, the distribution of individual attributes or the conditioning of other attributes. Sometimes, the domain experts would browse through the mined rules and search for evidence that could support, complement or reject the hypotheses flashing in their mind.

Users’ judgement of surprisingness is also subject to the quality/quantity of the information provided by rules. Occasionally, the domain experts suggested it was hard to judge the surprisingness of a rule; instead, the rule was regarded as meaningless. This case might indicate that rules have to be meaningful before the experts could evaluate surprisingness; the meaningfulness issue is further justified in Chapter 4. Another situation in our experience is that the experts might be unaware of some knowledge that they possess at first glance of a rule, but if additional assistance is provided then the experts could recall the relevant knowledge and form another opinion about the rule.

These two situations suggest that merely ranking mined rules, the conventional scope of unexpectedness, is insufficient to fully address the surprisingness problem. In fact, mining surprising patterns has at least three facets of concern: the meaningfulness of rules, methodologies for determining unexpectedness, and assistance for pattern interpretation.
1.2. **SCOPE**

Because data mining is a broad field with many different types of tasks for different kinds of data, e.g. classification, clustering, association for tabular data, relational data, text data, it is worth clarifying the precise scope of our investigation. We focus on association rule mining and its derivatives (see the introduction in Appendix B); hence the data we work with is in tabular format. Although temporal mining is important in many domains and temporal data is part of the datasets we explore in this thesis, we do not exploit the temporal interpretation of the data.

A complete KDD cycle generally consists of five steps: data selection, preprocessing, transformation, data mining, and pattern interpretation (Fayyad et al., 1996). Various kinds of knowledge have to be communicated and applied between domain experts and data analysts throughout the KDD cycle. In the thesis, we specifically concentrate on the usage of domain knowledge in the steps after data transformation, that is data mining (including post-processing) and pattern interpretation. Figure 1.2 shows the position of the scope of the thesis in the cycle of question formation and KDD.
1.2.1 Domain Knowledge

Much existing research attempts to incorporate knowledge into data mining for various purposes, e.g. for speeding up DM by reducing the search space; for improving the accuracy of classification; for highlighting interesting patterns; or for automating the KDD cycles. Whilst the term ‘knowledge’, in previous research, broadly refers to the knowledge about data mining and the problem domain, most research focuses on finding ways to make use of the domain knowledge of experts. We find it is difficult to induce a single meaning for the term ‘domain knowledge’ from previous research, because it refers to multiple facets of knowledge. In earlier work, the term ‘domain knowledge’, sometimes called ‘background knowledge’ or ‘prior knowledge’, has three meanings: (1) the knowledge of the domain (in all formats), (2) the knowledge in the domain experts’ heads, and (3) the encoded machine readable knowledge. We will use domain knowledge (DK) to indicate the knowledge in a user’s head, and use knowledgebase (KB) to denote encoded machine readable knowledge.

1.2.2 Unexpectedness

‘Unexpected’ is a term that, in data mining, generally refers to a computational property of rules with respect to a formal, computable knowledgebase. That is, unexpectedness is a mathematical metric applied to mined rules or patterns, but is a derivative of both a specific dataset and a particular formal representation of user knowledge (KB) related to that dataset. The value of unexpectedness will change if the KB changes.

If we agree that surprising facts are likely to be ‘interesting’ to the user - that is, new facts are most likely to be interesting - then it is reasonable to suggest that unexpectedness metrics should be better approximations of ‘interesting’ than objective interestingness metrics, purely because they can take some approximation of the user’s knowledge into account when ranking facts. Some researchers have also used ‘unexpected’ as a characteristic of a user’s impression or assessment of a fact or rule; however, in order to avoid confusion, we will use the terms unexpected and unexpectedness only in reference to the computational properties of knowledge.

Towards the original goal of KDD, the most desirable data mining result is that the mined rules are meaningful, interesting and surprising. However, due to the complexity of each problem, we focus on the goal of finding surprising
patterns/rules for users – regardless of whether they are actually interesting or not to them. As for meaningfulness, we research it separately from unexpectedness (see Chapter 4), and the solution could be a building block toward unexpectedness.

1.2.3 

**Surprise**

The fields of cognitive science and symbolic artificial intelligence (AI) have studied surprise in its fundamental causes, formal models, and its role in belief change (Meyer et al., 1991; Macedo and Cardoso, 2001; Lorini and Castelfranchi, 2006, 2007). Lorini and Castelfranchi (2007) classified two kinds of surprise – mismatch-based surprise (due to a recognised inconsistency between an expectation under scrutiny and a perceived fact), and astonishment (due to the recognition of the implausibility of the perceived fact, where this recognition is based either on the retrieval of a background expectation or on some inferential process). Under the categorisation, the term ‘surprising patterns’ in our research should refer to mismatch-based surprise where the pattern mismatches the users’ scrutinized expectation (Lorini and Castelfranchi, 2007), i.e. the expectation on which the user is focusing his/her attention and that the user tries to match with the pattern.

Surprise is suggested to be perhaps the most important causal precursor of belief change (Lorini and Castelfranchi, 2007). Studies of BDI agents (Rao and Georgeff, 1991; Subagdja et al., 2009) have formalised the functional role of surprise with respect to resource bounded belief revision (Lorini and Castelfranchi, 2006). The mental process and symbolic AI view of surprise can be a theocratical motive for belief update in designing the paradigm of unexpectedness.

1.2.4 Knowledge Acquisition

There is a body of research dealing with unexpectedness (as reviewed in Subsection 2.4), and most of the previous research focuses on fundamental problems such as knowledge representation and the calculation of unexpectedness.

---

4The terms ‘belief’, ‘expectation’ and ‘knowledge’ are treated differently in cognitive science, but were made synonyms by several authors of unexpectedness.
On top of the previous research, there is a question to be addressed – what problems would arise in terms of usability when carrying out a model of previous research? From a user’s perspective, the problems might exist in: (a) the overhead for learning the language of a knowledge representation, (b) the efforts of putting knowledge into a KB, (c) the energies in interpreting unexpected patterns, (d) more efforts in updating the KB, and more. Therefore the design of knowledge acquisition plays a critical role in the usability of the KDD system.

There is, of course, a tremendous amount of knowledge in a user’s head. However, what knowledge is needed for the computation of unexpectedness? Actually, we cannot know for sure because we do not know how the brain performs reasoning. In opposite, the general approach to unexpectedness is to devise a knowledge representation and equations to calculate unexpectedness. This approach is based on the assumption that the knowledge representation is capable of capturing the essential knowledge for unexpectedness, and the formulated equations can approximate the way of human reasoning.

In fact, any knowledge representation is limited in its expressibility, so that only a fraction of user’s knowledge could be captured. This implies that a great chunk of knowledge that might be involved in the reasoning process of unexpectedness would be left out of the KB; consequently, the calculation of unexpectedness is only an approximation.

The design of knowledge acquisition of unexpectedness faces a trade-off between completeness and convenience. Recognising the theoretical limitation aforementioned, we propose that it is more practical to formulate the knowledge acquisition process in an efficient and effective way, so that the acquisition overhead is low and it still can find surprising rules.

1.3 Research Question

From the users’ perspective, patterns derived during the data mining process are not always interesting. Mining unexpected patterns is a computational technique in earlier work addressing this problem. However, ‘unexpected patterns’ are not necessarily surprising to the user. Towards the goal of discovering new knowledge for KDD (knowledge discovery and data mining) users, we ask a seemingly simple question.
Research Question

How to mine surprising patterns for a KDD user?

This research question consists of various sub-questions that unfolded during the course of the research; they are enumerated in the list below.

Knowledge Representation. What knowledge representation should be used? The choice of knowledge representation is subject to three factors: (1) the capability of the representation, (2) the learning overhead of the user, and (3) the communication overhead during knowledge acquisition. Ideally, richer capability and lower overhead during learning and inputting is desirable. The overhead issues lead to the next problem.

Acquisition Interface. What if the user is not familiar with the knowledge representation? An ideal design of knowledge acquisition should not require the user to master the knowledge representation; instead, intuitive guidance and support should be deployed.

Meaningfulness. What if the user thinks the rule is meaningless? Meaningless rules are unwanted by users; in our experience, the user could not comment on a rule’s surprisingness if the rule was regarded as meaningless.

Interpretation Assistance. What if the user needs to know why the rule is valid when the rule contradicts the user’s knowledge? Association rules and their generics only provide limited statistical measures of rules. However, when a rule contradicts the user’s knowledge, the provided statistical measures are usually insufficient. Therefore, more information is needed to assist the user to understand the rule.

These sub-questions are addressed respectively in Chapters 2, 4, 5 and 6; we further discuss the sub-questions of the research question in Section 1.5 where we present the thesis outline.

1.4 Thesis Contribution

Finding surprising patterns is a desirable function for KDD systems, especially for those with the association rule mining function. Unexpectedness has been
researched for over a decade, yet it is still in its early stage where neither academic research nor commercial software have come up with well-accepted and widely deployed solutions; instead, there are new methods proposed from time to time. The contribution of the thesis is to enrich the research of unexpectedness from the viewpoint of pragmatically communicating with users. Based on this viewpoint, the thesis compares the existing approaches of unexpectedness and proposes principles to evaluate them; addresses the meaningfulness problem with the use of ontologies; augments the elements of unexpectedness based on Bayesian networks; presents a novel approach for explaining unexpected rules; reports case studies of real-world data by working closely with domain experts; finally, logically analyses the case studies to gain new understanding of surprisingness. The detailed contributions are described below.

- New building blocks for unexpectedness. Based on the knowledge representation of Bayesian networks, also used by Jaroszewicz and Simovici (2004), we propose new designs with new building blocks, such as a statistical independence-test for association rules, and network localisation for scalability. In terms of usability, we propose the use of association graphs as the bridge to build up Bayesian networks.

- A novel approach for assisting pattern interpretation, i.e finding explanations. We present a method to find explanations for unexpected rules; the method is based on probabilistic dependencies. In the approach, relevant variables are selected from rules and from other data sources to facilitate human understandable interpretations.

- A comprehensive review of unexpectedness. We thoroughly analyse existing approaches to unexpectedness from the aspects of knowledge representation and knowledge acquisition. We also propose three general principles for designing unexpectedness systems.

- Ontology-driven data mining. We investigate the use of openly available ontologies as a knowledgebase to deal with the problems of meaningfulness and unexpectedness. The experiment demonstrates that the use of a medical ontology in specifying the semantic structures of association rules could uncover semantically meaningful and expected rules of the user. In terms of the unexpectedness problem, we recommend the indi-
individual level relations are the essentials to utilise an ontology in mining unexpected patterns.

• Case studies by working closely with domain experts. We carry out experiments in a medical and an educational domain. In the medical domain, we analyse the data of chronic kidney disease patients with a nephrologist; in the educational domain, we analyse the data of a decimal comparison test of children with two education researchers. From the case studies, we learn the capabilities and the limitations of the proposed solutions, and acquired a deeper understanding of the essentials of mining surprising patterns.

• Suggestion of extending the scope of unexpectedness. The lesson learned from the case studies indicates that there are more gaps of mining surprising patterns that are not in the scope of unexpectedness; instead, the gaps happen in every step of the KDD. Therefore, the scope of mining surprising patterns should be extended to the whole process of KDD.

1.5 Thesis Outline

The thesis now proceeds as follows:

• Chapter 2 presents the necessary background material on the topics covered in the thesis, including a brief list of knowledge-rich data mining methodologies, an overview of subjective interestingness, a justification of the nomenclature of interestingness, a comprehensive review for unexpectedness which aims to find surprising patterns, a brief review for explanation generation for association rules, a review of knowledge acquisition, and a concise overview of ontology in data mining.

• Chapter 3 introduces the data used in the case studies, and the arrangements of how the data will be used.

• Chapter 4 introduces two designs which apply domain ontologies to the mining of association rules and unexpected rules. The first design provides a means to mine semantically meaningful association rules in responding to the problem of the meaningfulness of the research question. The second design hypothetically outlines a way to mine unexpected patterns based on an ontology base.
Chapter 5 develops the core design for unexpected pattern mining, where the ultimate goal is finding surprising patterns. In the chapter, we identify possible problems in unexpected pattern mining from the aspect of knowledge acquisition from users; and then present solutions for the identified problems. By combining these solutions, we propose an unexpectedness design where Bayesian networks serve as the knowledge representation.

Chapter 6 presents a novel idea in assisting pattern interpretation. The idea is to find possible explanations for unexpected rules in a probabilistic manner; and it could be applied to the knowledge aggregation of unexpectedness.

Chapter 7 empirically tests the proposed design for finding surprising patterns in two domains by using users who are domain experts. The first experiment is in the medical domain; the dataset comes from a registry of chronic kidney disease patients. The second experiment is in the education domain; the dataset is about children’s misconception of decimal notation. The results are critically analysed and learned lessons are reported.

Chapter 8 reviews the whole thesis against the research question to highlight the achievements and questions for future research. Finally, we conclude with a summary of the overall research and suggest new directions for mining surprising patterns.
Chapter 2

Knowledge Rich Data Mining

This chapter covers general background material for the thesis and provides comprehensive reviews of four related topics that are investigated in the thesis. In the beginning, we broadly examine the various approaches of knowledge rich data mining (Section 2.1). In the second part, a concise review of ontology in data mining is reported (Section 2.7). Then we present a comprehensive review of unexpectedness and propose a criterion to evaluate different approaches (Section 2.4). Thirdly, we report a brief review of the topic explanation generation (Section 2.5). Finally, because the knowledge for data mining has to be acquired somehow, and it is especially important for unexpectedness, a short review of knowledge acquisition is presented (Section 2.6).

2.1 Overview

There are many usages of domain knowledge in existing research in response to various limitations of data-oriented data mining. We briefly enumerate some of them (not a complete list) to provide an overview for knowledge rich data mining.

**Automated Knowledge Discovery.** Automated knowledge discovery aims to automate the scientific process, i.e. to carry out cycles of scientific experimentation with little human intervention. King et al. (2004) implemented a robot scientist ‘who’ continuously generates hypotheses ⊕ performs experiments for modeling an aromatic amino acid pathway in yeast. The core of machine learning in their study is inductive logic
programming (ILP), which can utilize background/prior knowledge\textsuperscript{1} and perform reasoning. Their system automatically originates hypotheses to explain observations, devises experiments to test these hypotheses, physically runs the experiments using a laboratory robot, interprets the results to falsify hypotheses inconsistent with the data, and then repeats the cycle. King et al. (2004) developed a logical formalism to model a cellular metabolism that captures the key relationship between the protein-coding sequence. All objects and relationships are described as logical formulae. Furthermore, directed graphs and undirected graphs are used for metabolic pathways and reactions respectively.

Secondly, Livingston et al. (2003) and Buchanan and Livingston (2004) presented an agenda- and justification- based framework, and demonstrated its application in protein crystallography and clinical data of patients in rehabilitation for a medical disability. The proposed method (HAMB) is an agenda-based heuristic search program that searches a space of possibly interesting items and associations. There are two kinds of knowledge used by HAMB: (i) domain-independent heuristics and (ii) domain-specific information. The domain-specific information is used in: (a) reducing redundancy by eliminating synonyms, (b) reducing the number of uninteresting discoveries and (c) reducing the number of non-novel discoveries. On the other hand, the domain-independent heuristics are used for (a) performing tasks, (b) evaluating the results of performing a task, (c) proposing new tasks, (d) assigning reasons and corresponding strengths for a proposed task, and (e) estimating the interestingness of the items.

Constructive Induction. Constructive induction (Michalski, 1983) is a well studied technique which transforms data into new representation space for mining more meaningful and accurate knowledge. It is also known as ‘new term problem’, ‘feature extraction’ and ‘feature generation’. Generating new features may involve different constructive induction strategies; some popular strategies are hypothesis-driven, data-driven

\textsuperscript{1}In the thesis, ‘knowledgebase’ is a term that may refer to background or prior knowledge. But in the description of earlier work, we preserve the common term ‘background knowledge’.
and knowledge-driven (Lo and Famili, 1997). Lo and Famili (1997) suggested two common types of domain knowledge:

- **Process or System Knowledge** that represents various aspects of domain theory. This type of knowledge comes from domain experts, empirical results, system design information, and system documentation.

- **Data Characteristics Knowledge** that is acquired during data preprocessing. Various approaches are taken. Examples are: data visualization, principal component analysis, dimensional analysis and data fusion. The main goal is that by understanding the nature of the data (combined with domain theory), one can generate new attributes.

The fusion of domain knowledge into constructive induction has been studied by Srinivasan and King (1999); Kudenko (2000) and others. Srinivasan and King (1999) examined a role of inductive logic programming (ILP) programs in extracting potentially useful attributes from problem-specific background knowledge, which is engineered in Prolog language. Kudenko (2000) used an ontology as background knowledge for feature generation.

**Constrained Clustering.** Clustering, traditionally viewed as an unsupervised learning, can benefit from additional knowledge about the problem domain if the knowledge is available (Wagstaff et al., 2001). Two types of constraints (knowledge) are considered in the research of (Wagstaff et al., 2001): must-link and cannot-link, that specify two instances that should or should not be placed in the same cluster. Klein et al. (2002) proposed making use of domain knowledge for space-level constraints for addressing the limitation of pairwise instance constraints.

**Inductive Logic Programming.** Different from other propositional data mining algorithms, inductive logic programming (ILP) is capable of mining relational data. Briefly, ILP performs inductive construction of first-order clausal theories from examples and background knowledge (Muggleton and Raedt, 1994). Bryant (1997) used Progol, a domain independent ILP tool which is available in the public domain, to perform data mining on a chemical database. Džeroski (2003, 2006) introduced how to
mine patterns that involve multiple tables (relations) from a relational database. The proposed method is composed of transforming ILP problems to propositional form, upgrading propositional approaches, finding relational association rules and relational decision trees. In terms of temporal concepts, Ichise and Numao (2003) introduced a temporal relationship mining algorithm which applies temporal predicates in ILP.

**Subjective Interestingness.** Subjective interestingness (further discussed in Section 2.2) incorporates users’ knowledge into the ranking of rules. Here we briefly introduce some related work which aims to search for a best classifier among classification rules. Li et al. (2009) applied concept hierarchy (taxonomy) to a novel interestingness metric where the ‘importance’ of rules can be calculated according to users’ preference. The approach is facilitated in two stages. Firstly, the user assign a concept hierarchy for the given data so that the attributes of data can be linked under corresponding concepts; next, the user assigns an ‘importance’ weight to each concept in the taxonomy. In the second stage, the prepared concept hierarchy and ‘importance’ weights $w_c(k)$ are applied to mined rules to calculate their ERIM (Enhanced Rule Importance Measure):

$$ERIM_i = \sum_{p=1}^{l_i} w_c(k),p,$$

where $l_i$ is the number of attributes contained by rule $i$ and $w_c(k),p$ is the weight of the $p$th attribute in this rule.

Suzuki (2009) proposed an interestingness measure for groups of classification rules which are mutually related based on the Minimum Description Length Principle (MDLP). The approach incorporates users background knowledge into the calculation of the encoding length of classifiers. Let $D$ be the data and $B$ be the background knowledge of the user. A classifier model $M$ is obtained by the equation:

$$M \equiv \arg \min_M (-\log P(M) - \log P(D|M) - \log P(B|M)).$$

**More topics.** The topics on ontology, surprisingness and pattern interpretation, which are covered by this research, are discussed in detail in later sections.
2.2 Subjective Interestingness

There is a well recognized problem that data mining (especially association rules mining) often generates enormous numbers of patterns which usually are of no interest to a user (Silberschatz and Tuzhilin, 1996; Liu et al., 1999a). Aforementioned in Chapter 1, a general solution to this problem is to measure the interestingness of patterns so that the result can be pruned or ranked. Some studies have tackled this problem by defining interestingness in terms of statistical significance, such as support, confidence, Gini index, Lapalace correction (Clark and Boswell, 1991) and so on. Because these kinds of measures do not take a user’s knowledge into consideration, they are called objective interestingness measures (Geng and Hamilton, 2006; McGarry, 2005). Ohsaki et al. (2007) tested 40 different objective interestingness in mining clinical data, and found that some measures are useful in estimating the interest of medical experts. It is suggested by Ohsaki et al. (2007) that using flexible combinations of objective measures via a user interface for post-processing would be more useful than a framework where a mining algorithm fixes on a few prescribed objective measures in its mining process.

Some researchers have observed that the objective measures of interestingness, although useful in many aspects, usually do not capture all the complexities of the pattern discovery process (Silberschatz and Tuzhilin, 1996), and the users’ knowledge should be considered. Consequently, various subjective measures have been proposed to incorporate the knowledge of a user. The most studied subjective measure is unexpectedness, that a pattern is unexpected if it conflicts with a person’s existing knowledge or expectations (Geng and Hamilton, 2006).

Figure 2.1 compares the paradigms of objective and subjective interestingness. Figure 2.1(a1) depicts conventional data mining algorithms which use simple interestingness measures, such as support in frequent itemsets (Agrawal et al., 1993). The paradigm in Figure 2.1(a2) shows an additional data-driven interestingness filter which processes the patterns generated from basic data mining algorithms, sometimes called post-processing, e.g. (Guillaume et al., 1998). On the other hand, subjective interestingness, Figure 2.1(b), utilises users’ knowledge to gauge the interestingness of patterns.

The thesis follows previous research (will be reviewed in Section 2.4) that unexpected patterns should be mined by utilising users’ knowledge in its
CHAPTER 2. KNOWLEDGE RICH DATA MINING

Figure 2.1: Comparison between objective and subjective interestingness. (a1,a2) Two paradigms of objective interestingness. (b) Paradigm of subjective interestingness.

computation (thus it belongs to subjective interestingness). Next subsection presents a nomenclature of subjective interestingness to pin down concepts, e.g. unexpected/surprising, discussed in the thesis.

2.2.1 Justification of Terminology

One trivial fact is that computationally unexpected rules are not always truly surprising to users. Therefore, we propose a nomenclature to denote the concepts of surprisingness/unexpectedness in respect to users and KDD systems (Figure 2.2). The terms in the terminology are indexed in page 191.

The knowledge needed for subjective interestingness is domain specific knowledge about the data, therefore it is called domain knowledge. Clearly, a domain expert will have much more specific domain knowledge than a non-expert, and any newly presented fact/pattern/rule will be assessed as interesting or not with respect to the knowledge that a user already has. So, before we can identify potentially interesting facts/patterns/rules, we need to know what the user knows. Here we define domain knowledge (DK) as the

---

2 The direction of an arrow represents a subject-object structure of a sentence describing the elements; for example, the [User] [thinks a rule] is [Interesting].
2.2. SUBJECTIVE INTERESTINGNESS

Figure 2.2: The nomenclature of interestingness used in the thesis.

set of knowledge of an individual user interacting with a data mining system. Domain knowledge refers to terms, concepts, relationships and facts relevant to the dataset being mined; it will vary from user to user, approaching zero in naïve users. Further, to automate the process of discovering interesting facts/patterns/rules, we need to record some part of the domain knowledge of the user in a computable form, which we refer to as the knowledgebase (KB).

Unexpectedness was originally defined over the contradiction between the pattern and the user’s expectations\(^3\) (Silberschatz and Tuzhilin, 1996). Because ‘contradiction’ is a specific term in logics, we reserve it for referring to the computational quality of patterns/rules; thus, a pattern/rule might be

\(^3\)For being consistent with the data-mining literature, we treat expectations or beliefs as synonyms of knowledge; although many of religious persuasion may argue that beliefs and knowledge are not the same thing at all; furthermore, there are formal philosophical logic treatments of belief and knowledge that provide an objective separation.
contradictory to the KB. We also use conflicting as the counterpart of ‘contradictory’ in the aspect of human judgement; that is, a pattern/rule might be conflicting to the DK.

Unknown is another sufficient condition for unexpectedness (Liu et al., 2000). There are occasions when the mined rule does not match any clause in the KB, and we cannot derive any related clause from the KB which can show contradiction; a trivial example of the case is that the KB is empty. We use the term ‘unrecognised’ to denote this situation; that is, a rule might be unrecognised by the KB. Because we research a probabilistic approach for unexpectedness, which in its nature does not separate unrecognised from contradictory (justified in Subsection 5.3.5), we classify unrecognised as one kind of unexpected.

Note that previous research has used the term ‘novelty’ to denote the characteristic of a rule, in which the rule should be unknown to a user. From the computational point of view, ‘unrecognised’ would be a better term than ‘novel’ in describing the process of reasoning a rule by a KB.

In total, we use surprising as a subjective user-perspective term, that defines the set of facts that is true, but conflicting (that is, conflicting with a user’s knowledge) or unknown to the user. An alternative adjective to surprising might be unexpected; because it is the term commonly used in the research of KDD, we reserve ‘unexpected’ for computational assessment of patterns. Examples of surprising facts might be: DK={‘all swans are black’}; conflicting fact={‘some swans are white’}. Sometimes, rules could be unknown (to a user’s knowledge); and this is a kind of interesting rules as well. Examples of unknown facts might be: DK={‘all swans are black’}; unknown fact={‘some swans cannot fly’}. Note that the ‘status’ of a fact can change between unknown and conflicting according to the set of knowledge against which it is being assessed. An example of this might be: DK={‘all swans are black’, ‘all swans can fly’}; conflicting fact={‘some black swans cannot fly’}. Meanwhile, the ‘status’ of surprising and unknown is not mutually exclusive; as we can imagine facts that are surprising and unknown - for example {‘black swans can talk’}.

To summarise, we can display the logic of how we approach the mining of interesting patterns by using this nomenclature. Start from finding interesting patterns, plenty of interestingness measures are developed either objectively or subjectively; because objective interestingness do not capture all the complex-
ities of the pattern discovery process (Silberschatz and Tuzhilin, 1996), subjective interestingness is studied for better results. Subjective interestingness, however, is still a broad topic in which to pin down an exact formula; therefore, unexpectedness is proposed as the most relevant condition for subjective interestingness. Unexpectedness could be further broken up into contradictory or unrecognized to the KB; therefore, the KB, which represents the DK, is critical for the success of unexpectedness. The ultimate aim of interestingness in KDD is to find interesting rules to users; previous research suggests that surprising rules are interesting; from the user’s perspective, surprising rules are conflicting or unknown to his/her domain knowledge. The aim of the thesis follows previous research in mining surprising (to a user) rules; in practice this goal has to be carried out in mining unexpected (to a system) rules.

2.3 Data-Driven Surprisingness

Existing research has proposed both objective (data-driven) and subjective (user-driven) approaches to surprising pattern mining. Exception rules and Simpson’s paradox are examples of objective approaches (Subsections 2.3.1 and 2.3.2). The merit but also the drawback of objective methods is that users’ knowledge is not taken into consideration – although no knowledge acquisition is required, these methods could not customise patterns for different users based on their knowledge. Subsections 2.3.1 and 2.3.2 briefly discuss data-driven approaches for surprisingness – exception rules and Simpson’s paradox. We leave the review of the subjective (user-driven) approach of surprisingness – unexpectedness – to the next section.

2.3.1 Exception Rules

Exception rules (ERs), although rarely compared in the literature of unexpectedness, share similarity with unexpected rules. Exception rules are devised based on the contrasting phenomenon between a ‘common sense’ rule and its subset which is stratified by another attribute (Suzuki and Kodratoff, 1998; Suzuki and Kodratoff, 1998).

\[\text{\footnotesize \cite{Suzuki1998}}\]

Although we intend to reserve the term ‘surprising’ for users’ judgement, this section exceptionally uses it to refer to the computational quality of rules for being consistent with earlier work.
Liu et al., 1999b; Hussain et al., 2000; Suzuki and Zytkow, 2005). Exception rules are claimed to possess the ability to identify surprising rules via proper statistical significance metrics.

Briefly, exception rules are rules which represent a deviation from strong rules (Suzuki and Kodratoff, 1998). Strong rules are rules with high support and high confidence, and they are named common sense rules based on the assumption that these rules should be already known by users. Given a common sense rule of binary attributes

\[ A \rightarrow X, \]

an exception rule consists of additional attribute(s) \( B \) that can reverse the prevalence of \( X \) under the condition \( A \), i.e.

\[ A, B \rightarrow \neg X. \]

The existence of the rule \( B \rightarrow X \) also needs to be taken into consideration in determining how exceptional the ER is; it is called reference rule and is expressed in the format of negative association:

\[ B \rightarrow \neg X. \]

Suzuki and Zytkow (2005) generalised the conventional definition of ERs above to one special case of 11 types of ERs. The 11 categories of ERs are derived from the combination and permutation of the attributes in the common sense rule, exception rule and reference rule. Let \( x, y, z \) be items or itemsets, Figure 2.3 depicts the 11 types of ERs.

Hussain et al. (2000) asserted a debate to promote the merit of objective interestingness:

"It is true that interestingness is a relative issue that depends on the other prior knowledge. However, this estimation can be biased due to the incomplete or inaccurate knowledge about the domain. Even if possible to estimate interestingness, it is not so trivial to judge the interestingness from a huge set of mined rules. Therefore, an automated system is required that can exploit the knowledge extracted from the data in measuring interestingness. Since the extracted knowledge comes from the data, so it is possible to find a measure that is unbiased from the user’s own belief. An unbiased
2.3. DATA-DRIVEN SURPRISINGNESS

Figure 2.3: The categories of exception rules (Suzuki and Zytkow, 2005).

A measure that can estimate the interestingness of a rule with respect to the extracted rules can be more acceptable to the user.

According to Hussain et al. (2000), the unbiased common sense rules, derived from data, should be applied to extracting rules which are more acceptable to the user, because individual knowledge might be biased due to the incomplete or inaccurate knowledge about the domain.

We disagree with the argument of (Hussain et al., 2000) for one reason – the incompleteness or inaccuracy of an individual’s knowledge is in fact the motive of KDD. Suppose a medical dataset incorporated the fact \( \text{kidney dialysis} \rightarrow \text{heart attack} \) with high support and high confidence. This fact, in the definition of ERs, is a common sense rule which the user should have known it; however, it could be surprising to a naïve user who has little medical knowledge.

Duval et al. (2007) further extended the scope of ERs to subjective approaches of unexpectedness, e.g. the work by Silberschatz and Tuzhilin (1995); Padmanabhan and Tuzhilin (1998, 1999). This broadened view of ERs maintains the opinion that the common sense knowledge, which is used in ERs,
could be either automatically learned from data or provided by the user.

In total, the earlier work on ERs mainly tackles the unexpectedness problem from data-driven approaches (Duval et al., 2007). However, data-driven approaches do not maintain a KB which can reflect the difference between individuals’ knowledge which we regard as being important in the mining of surprising patterns on a per-user basis. Therefore, the thesis researches on subjective (user-driven) approaches to the research question.

### 2.3.2 Simpson’s Paradox

Simpson’s paradox (Simpson, 1951), well known in statistics, has been used as a criterion for detecting unexpected/surprising patterns in data-driven approaches (Fabris and Freitas, 1999, 2006; Freitas et al., 2007). Let the event $C$ be the apparent ‘cause’ of an event $E$, the ‘effect’. Simpson’s paradox occurs if the event $C$ increases the probability of the event $E$ in a given population and, at the same time, decreases the probability of event $E$ in every subpopulation which is conditioned by a confounding variable $F$ (Pearl, 2000). The mathematical definition of Simpson’s paradox is characterised by the following three inequalities:

\[
P(E|C) > P(E|\neg C) \\
P(E|C, F) < P(E|\neg C, F) \\
P(E|C, \neg F) < P(E|\neg C, \neg F)
\]

Consider the hypothetical example about drug effectiveness provided by (Pearl, 2000, chap. 6). Table 2.1 shows the recovering rate of a drug ($C$) in the entire population; the outcomes may suggest that the drug $C$ improves the recovery rate. However if we separately inspect the outcomes in the subpopulations of males and females (Table 2.2), it suggests that receiving the drug reduces the recovery rate – a totally reversed conclusion.

<table>
<thead>
<tr>
<th>Recovery</th>
<th>Combined</th>
<th>$E$</th>
<th>$\neg E$</th>
<th>Total</th>
<th>Recovery rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Drug</td>
<td>$C$</td>
<td>20</td>
<td>20</td>
<td>40</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>$\neg C$</td>
<td>16</td>
<td>24</td>
<td>40</td>
<td>40%</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>36</td>
<td>44</td>
<td>80</td>
<td></td>
</tr>
</tbody>
</table>
Table 2.2: Recovery rates for the sub-populations of males and females separately.

<table>
<thead>
<tr>
<th>Recovery</th>
<th></th>
<th></th>
<th>Total</th>
<th>Recovery rate</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Males</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Drug</td>
<td>C</td>
<td>18</td>
<td>12</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>¬C</td>
<td>7</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Total</td>
<td>25</td>
<td>15</td>
<td>40</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Recovery</th>
<th></th>
<th></th>
<th>Total</th>
<th>Recovery rate</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Females</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Drug</td>
<td>C</td>
<td>2</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>¬C</td>
<td>9</td>
<td>21</td>
<td>30</td>
</tr>
<tr>
<td>Total</td>
<td>11</td>
<td>29</td>
<td>40</td>
<td></td>
</tr>
</tbody>
</table>

Again, Simpson’s paradox, which is a data-driven approach for surprisingness, does not maintain a KB which can reflect the difference between individuals’ knowledge, which we regard as being important in the mining of surprising patterns on a per-user basis. The thesis will not use it in our solutions.

2.4 Review of Unexpectedness

Unexpectedness is a kind of subjective measure which aims to find surprising patterns. In comparison to other topics in data mining, unexpectedness has drawn comparatively little attention in the last decade. One reason might be that unexpectedness requires the element of a user’s knowledge; however, acquiring a user’s knowledge is a non-trivial problem, i.e. the problem of knowledge acquisition.

Abstractly, unexpectedness is the difference between expectation and observation:

\[ \text{Degree(Unexpectedness)} = \text{Difference(Expectation, Observation)}, \quad (2.1) \]

and we will see that all related work used this definition explicitly or implicitly.

One of the difficulties in mining unexpected patterns comes from the innate character of entangled interaction between the user and the data mining algorithm; let us explain it by the general routine of finding unexpected patterns. At the beginning, KB has to be acquired somehow. Second, the mining algorithm calculates the degree of unexpectedness for the mined patterns
against the KB and presents results to a user. Third, the user learns some knowledge from the unexpected patterns. Finally, the user refines the KB and performs the next iteration from the second step. Therefore, the user’s knowledge evolves dynamically as well as the KB. The dynamic feature of mining unexpected patterns introduces many issues that are different from those of common one-off data mining tasks. For example, how to acquire KB efficiently and utilize the KB intelligently are two key factors for providing smooth interactions for users. Figure 2.4 depicts the knowledge dynamics discussed above.

We review the earlier work of unexpectedness from the aspect of knowledge representations (Subsection 2.4.1) and calculations of unexpectedness (Subsection 2.4.2). We then review unexpected pattern mining from the aspect of knowledge acquisition and knowledge update in Subsection 2.4.3. Based on the review, we further analyse the limits and capabilities of various types of knowledge representation that were used in previous research in Subsection 2.4.5. Based on the analysis, we propose three principles for unexpectedness mining systems (Subsection 2.4.6) and sum up the review in Subsection 2.4.7.

2.4.1 Knowledge Representation Techniques

This subsection reports on approaches to representing a user’s knowledge (DK) in unexpectedness. Briefly, earlier work has used rules, fuzzy rules, probabilities, Bayesian networks, and fuzzy taxonomy to represent DK.

Liu et al. (1997) proposed a technique for analyzing discovered rules in the context of users’ fuzzy expectations, called general impressions, to help
a user identify interesting rules. The meaning of general impressions is that a user’s domain knowledge is not available as a set of detailed, explicit concepts and relationships, but rather as a set of abstract ‘feelings.’ General impressions are represented by rules involving impression descriptors: $ID \in \{\langle,\rangle,\ll,,|,[aset]\}$, which describe the impression over attributes. The form of general impressions (GIs) is a combination of attributes with $ID$ and likely class. Some meanings of GIs are illustrated as follows:

- $a \rightarrow C_j$ This represents the impression that a bigger value of $a$ will result in a higher likelihood of being in class $C_j$.
- $a \ll \rightarrow C_j$ This represents the impression that if the value of $a$ is within some range, then class $C_j$ is likely to be the result.
- $a| \rightarrow C_{sub}$ This represents the impression that there exist some relationships between the attribute $a$ and the classes in $C_{sub}$ where $C_{sub} = \{C_1,\cdots,C_f\}$. However, the exact relationships are not known.

The format of general impressions (GIs) is similar to classification rules. The example below represents an expert’s impression about loan screening criteria; for instance, if an applicant has big savings, even though he has no job, the loan is likely to be approved. Below is a set of possible GIs of the example:

1. saving $\rightarrow$ approved,
2. age $| \rightarrow \{\text{approved, not approved}\}$,
3. jobless $\{\text{no}\} \rightarrow$ approved,
4. jobless $\{\text{yes}\} \rightarrow$ not approved,
5. saving $\rangle$, age $\ll \rightarrow$ approved,
6. saving $\rangle$, jobless $\{\text{yes}\} \rightarrow$ approved.

Liu et al. (1999a) introduced fuzzy linguistic variables for representing a user’s imprecise knowledge or hypotheses. A fuzzy linguist variable is a quintuple $(x,T(x),U,G,\tilde{M})$ in which $x$ is the name of the variable; $T(x)$ is the term...
set of $x$, with each value being a fuzzy variable denoted generally by $x$ and ranging over a universe of discourse $U$; $G$ is a syntactic rule for generating the name, $X$, of values of $x$; and $M$ is a semantic rule for associating with each value $X$ its meaning, $\tilde{M}(X)$ which is a fuzzy subset of $U$. A particular $X$ is called a term. As an example in the traffic-collision domain, if speed is interpreted as a linguistic variable with $U = [1,140]$, then its term set $T(speed)$ could be:

$$T(speed) = \{\text{slow}, \text{moderate}, \text{fast}, \ldots\}.$$  

With the fuzzy linguistic variables, a user's expectation is represented in the same format as discovered patterns: classification rules. For instance, a user-expected pattern would be:

**IF** Age=OLD, Location=BAD-VISIBILITY  
**THEN** Class=BAD-ACCIDENT

In comparison to (Liu et al., 1997), fuzzy linguistic variables allow users to define semantic meanings over the possible values of each attribute.

Continuing the study of (Liu et al., 1997), Liu et al. (2000) combined general impressions with reasonably precise concept (RPC) (Liu and Hsu, 1996) and precise knowledge (PK). The representation of general impressions is changed to a set of possibly related attributes without impression descriptors. RPC represents a user's concept that there should be some associations among some classes of items, and the user also knows the direction of the associations; it is expressed as:

$$rpc((S_1, \ldots, S_m \rightarrow V_1, \ldots, V_g)) [\text{support}, \text{confidence}],$$

where

1. $S_i$ or $V_i$ is either: an item, a class, or a subset of classes.

2. $RPC$ represents a complex disjunctive propositional formula.

3. $Support$ and $confidence$ are optional.

PK is based on the definition of RPC with more restrictions, e.g. the $support$ and $confidence$ need to be specified. Moreover, class hierarchy (or taxonomy) is used to describe the categorization of attributes, where the categorization could be used in $GI$, $RPC$ and $PK$. 

2.4. REVIEW OF UNEXPECTEDNESS

Wang et al. (2003) also used a similar representation to that in (Liu et al., 1999a); it is referred as knowledge rules. For example, here is a knowledge rule:

\[
\text{Education} = \text{Low} \rightarrow \text{Loan} = \text{No},
\]

where a fuzzy term Low is defined by a fuzzy set:

\[
\text{Low} = \{ (\text{Primary}, 1), (\text{Secondary}, 0.5) \}.
\]

The example shows that when an applicant’s education is low, defined as primary and secondary education, degree 1 and 0.5 respectively, the loan would not be approved. In addition, a preference model is introduced for specifying the user’s knowledge about how to apply knowledge rules to a given scenario or a tuple\(^6\). The preference model is specified by covering knowledge for each tuple. For a given tuple, the covering knowledge refers to one or more knowledge rules that match the tuple and the user prefers to apply to the tuple. There is also a covering depth\(=d\) refers to the number of knowledge rules in the covering knowledge, usually a small integer such as 1, 2, or 3. For example, if there is a set of covering knowledge \(\{K_1, K_2, \ldots, K_n\}\) about a tuple, and the covering depth\(=2\), then \(\{K_1, K_2\}\) will be used to calculate unexpectedness.

Shekar and Natarajan (2004) used fuzzy taxonomy to calculate ‘item-relatedness’ which is an indicator of unexpectedness. A fuzzy taxonomy is a tree with items represented as leaf nodes and concepts represented as non-leaf nodes; the links in a taxonomy usually represent ‘is-a’ relationships. The term ‘fuzzy’ here means that a ‘child’ node need not be a full member of the category/concept represented by its ‘parent’ node; a membership function between 0 and 1 is used in its calculation.

Padmanabhan and Tuzhilin (1998, 1999, 2002) defined expectations in the format of association rules expressed in logical literals. A belief has the form \(\text{body} \rightarrow \text{head} (X \rightarrow A)\) where \(X\) and \(A\) are conjunctions of literals (i.e., either atomic formulas of first-order logic or negations of atomic formulas). The definition is kept general and does not impose restrictions of the structures of atomic formulas that can appear in the literals of \(X\) and \(A\). For instance, a

\(^6\)Tuple refers to itemset in the context of association rules mining.
means professionals tend to shop more on weekends. Padmanabhan and Tuzhilin (2006) further studied *minimal unexpected patterns* as those unexpected patterns that are not refinements of other unexpected patterns. Notably, one advantage of logic-based representations is that they do not require the eliciting probabilities associated with beliefs from users, which is problematic in probabilistic approaches.

Silberschatz and Tuzhilin (1995, 1996) studied unexpectedness via defining beliefs (or expectations). Beliefs are defined as logical statements (predicate formula expressed in first-order logic) and are assigned some degree or measure, or a confidence factor to each belief. In particular, if $b$ is a belief based on some previous evidence $\xi$, then $d(b|\xi)$ denotes the degree of belief $b$. There are two types of beliefs: hard beliefs and soft beliefs. Hard beliefs are the constraints that cannot be changed with new evidence, and soft beliefs could be changed with new evidence. Silberschatz and Tuzhilin (1996) enumerated five approaches for computing the degree of soft beliefs, i.e. Bayesian, Dempster–Shafer, frequency, Cyc’s and statistical approach. Take the Bayesian approach as an example; the degree of a belief $\alpha$ is defined as the conditional probability, $P(\alpha|\xi)$, that $\alpha$ holds, given some previous evidence $\xi$ supporting that belief. Given new evidence $E$, the degree of belief in $\alpha$, $P(\alpha|E,\xi)$, is updated by the Bayes rule:

$$P(\alpha|E,\xi) = \frac{P(E|\alpha,\xi)P(\alpha|\xi)}{P(E|\alpha,\xi)P(\alpha|\xi) + P(E|\neg\alpha,\xi)P(\neg\alpha|\xi)}.$$ (2.2)

Among the five enumerated approaches, the Bayesian approach is suggested to be best suited for defining degrees of beliefs because it is the only appropriate approach to defining “degrees of plausibility” of logical statements satisfying certain intuitive desiderata (Silberschatz and Tuzhilin, 1996).

Jaroszewicz and Simovici (2004); Jaroszewicz and Scheffer (2005); Malhas and Aghbari (2009) presented the use of Bayesian networks (BN) for representing DK. Each variable is represented as a node in the network, and the edges are assigned manually so the user can express his/her belief of the dependencies among variables. Figure 2.5 shows an example of BN. Appendix C gives a basic definition of BN.
Xin et al. (2006) used log-linear and biased belief models to learn a user’s prior knowledge \( M \) from his/her interactive feedback. The mined rules are selected and presented to a user and the user provides ranked interestingness over the rules. The feedback is formulated as a constraint on the model to be learned. The constraint for pattern \( P_i \) is more interesting than \( P_j \), \( P_i > P_j \), is formulated as:

\[
R(f_o(P_i), f_e(P_i)) > R(f_o(P_j), f_e(P_j)),
\]

where \( R(\cdot) \) is the ranking function of interestingness; \( f_o \) and \( f_e \) denote observed and expected frequency respectively. There are two models for representing knowledge. One is a log-linear model which works for item-set patterns only, and the other is a biased belief model which can also be applied to structural patterns. The fully independent log-linear model for expected frequency is expressed as:

\[
\log f_e(P) = u + \sum_{j=1,...,s} u_j,
\]

(2.3)

where \( u \) is the interactions between items. In the biased belief model, the expected frequency \( f_e(P) \) is proportional to the expected number of occurrences of pattern \( P \) in Equation 2.4.

\[
f_e(P) \propto \sum_{k=1,...,m} p_k \times x_k(P),
\]

(2.4)

where \( x_k(P) = 1 \) if transaction \( k \) contains pattern \( P \), otherwise, it is 0.

To summarise, the knowledge representations can be categorized into three types:
Generic Rules - The general impressions (Liu et al., 1997), fuzzy rules (Liu et al., 1999a), reasonably precise concepts, precise knowledge (Liu et al., 2000), knowledge rules (Wang et al., 2003) and rules of beliefs (Padmanabhan and Tuzhilin, 2002, 1999, 1998) are rule based knowledge representations.

Probability - The Bayesian approach (Silberschatz and Tuzhilin, 1996, 1995), Bayesian networks (Jaroszewicz and Simovici, 2004; Malhas and Aghbari, 2009), log-linear and biased belief models (Xin et al., 2006) are based on probability.


2.4.2 Definitions and Calculations of Unexpectedness

This subsection focuses on the definitions and calculations of unexpectedness. The basic idea of unexpectedness is based on Equation 2.1.

Liu et al. (1997, 2000, 1999a) proposed approaches for matching and ranking discovered rules against a user’s expectation. The definition is: if a rule is similar to some predefined expectations, then this rule is expected; otherwise, it is unexpected. The matching function of (Liu et al., 1997) is defined as

\[
\text{match}(\text{rule}, KB) = \sum_{j=1}^{n} S\text{match}(\text{condition}(\text{rule}_j), KB_j).
\]  

Let \(a\) be an attribute, \(v\) is its value, \(OP\) is an operator, and \(ID\) is the impression descriptor, the \(S\text{match}\) function returns a discrete value in \(\{0, 0.2, 0.5, 1\}\) by a procedure called \(S\text{match}(aOPv, aID)\). Overall, the score of the match of a rule is aggregated from its comparison with all expectations.

In (Liu et al., 1999a), a fuzzy pattern matching is designed in two steps: attribute name match, and attribute value match. In the definition, \(E\) is the set of user-expected patterns, and \(B\) is the set of discovered patterns; \(W_i\) is the degree of match between \(B_i\) and \(E\). For rule \(i\) and expectation \(j\), \(W_{i,j}\) is computed from \(L_{i,j}\) (attribute name match), \(V_{(i,j)k}\) (\(k\)'th attribute value match), and \(Z_{i,j}\) (attribute value match of consequents). The attribute name
match is defined as

$$L_{(i,j)} = \frac{|A_{(i,j)}|}{\max(|B_i|, |E_j|)},$$  \hspace{1cm} (2.6)$$

where $A_{(i,j)}$ is the set of common attributes of $B_i$ and $E_j$. The attribute value matching functions, $V_{(i,j)k}$ and $Z_{(i,j)}$, are defined more complexly; please refer to the original paper for detail. In terms of unexpected pattern rankings, there are few types of unexpectedness. For example, *unexpected consequence* is defined as:

$$W_{(i,j)} = \begin{cases} 
L_{(i,j)} \times \sum_{k \in A_{(i,j)}} V_{(i,j)k} - (Z_{(i,j)} - 1) & |A_{(i,j)}| \neq 0 \\
-Z_{(i,j)} & |A_{(i,j)}| = 0.
\end{cases}$$

The technique of Liu et al. (2000) ‘matches’ and ranks mined rules with a user’s knowledge (GIs, RPCs and PKs) in a number of ways for finding different types of interesting rules: *conforming rules, unexpected consequent rules, unexpected condition rules and both-side unexpected rules*. In calculation, let $L_{ij}$ and $R_{ij}$ be the degrees of condition and consequent match of rule $A_{i}$ against $j$’th user’s belief $U_{j}$ respectively. The four types of interesting rules are computed as follows:

- **conm**

$$confm_{ij} = L_{ij} \times R_{ij},$$

- **unexpConseq**

$$unexpConseq_{ij} = \begin{cases} 
0 & L_{ij} - R_{ij} \leq 0 \\
L_{ij} - R_{ij} & L_{ij} - R_{ij} > 0
\end{cases}$$

- **unexpCond**

$$unexpCond_{ij} = \begin{cases} 
0 & R_{ij} - L_{ij} \leq 0 \\
R_{ij} - L_{ij} & R_{ij} - L_{ij} > 0
\end{cases}$$

- **bsUnexp**

$$bsUnexp_{i} = 1 - \max(Cfm_{i}, UCond_{i}, UConseq_{i}).$$

(2.7)

There are more definitions for computing the four types of interesting rules under the three types of users’ beliefs: GIs, RPCs and PKs. The overall concept is that a rule and a belief are ‘different’ if either the consequents of the rule and the belief are ‘similar’, but the antecedents are ‘far apart’ or vice versa (Padmanabhan and Tuzhilin, 1998).

Wang et al. (2003) defined unexpectedness by aggregating a rule’s violation with respect to the user’s knowledge:

$$vK(r) = agg\{v(r, K) | K \in K\},$$
where \( v() \) is the violation function of rule \( r \), and \( K \) is the knowledge rule in the covering knowledge \( \mathbb{K} \). The violation function over a single knowledge rule is defined as:

\[
v(r, K) = \begin{cases} 
    \overline{hm}(r, K) \times bm(r, K), & bm(r, K) \geq \sigma \land \overline{hm}(r, K) \geq \sigma \\
    0, & \text{otherwise.}
\end{cases}
\]

\( hm() \) and \( bm() \) are the match of head and body respectively. There are three unexpectedness measures defined over violation: unexpectedness, unexpectedness support and unexpectedness confidence. We omit the details here.

Silberschatz and Tuzhilin (1996, 1995) defined KB in probabilistic beliefs. The unexpectedness (originally called ‘interestingness’ in their paper) of pattern \( p \) to soft beliefs \( B \) and previous evidence \( \xi \) is defined as:

\[
I(p, B, \xi) = \sum_{\alpha_i \in B} w_i |d(\alpha_i|p, \xi) - d(\alpha_i|\xi)|,
\]

where \( d() \) is degree of belief and \( w_i \) is a weight of belief \( \alpha_i \). The interestingness of a pattern \( p \) relative to a belief system \( B \) and prior evidence \( \xi \) is defined in terms of how much degrees of beliefs change as a result of a new pattern \( p \).

Padmanabhan and Tuzhilin (1998, 1999, 2002, 2006) used a consistent definition of unexpectedness based on logical contradiction. A rule and a belief are defined as generalized association rules: \( A \rightarrow B \), where \( A \) and \( B \) are conjunctions of literals. Briefly, their definition is that a rule \( A \rightarrow B \) is unexpected with respect to a belief \( X \rightarrow Y \) on a dataset \( D \) if the following conditions hold.

1. \( B \land Y \models FALSE \). This condition states that \( B \) and \( Y \) logically contradict each other.

2. \( A \land X \) holds on a statistically large subset of tuples in \( D \). In other words, the proportion of \( A \land X \) in \( D \) should be higher than a threshold, e.g. support.

3. The rule \( A, X \rightarrow B \) holds. Since the first condition constrains \( B \) and \( Y \) to logically contradict each other, it follows that \( A, X \rightarrow \neg Y \) holds.

Shekar and Natarajan (2004) defined one kind of unexpectedness as ‘item-relatedness’. The expectation is defined as a user specified fuzzy taxonomy. The relatedness of two items is then calculated according to their highest-level
2.4. REVIEW OF UNEXPECTEDNESS

node membership $HM_{A,B}(p)$, highest-level relatedness $HR_{A,B}(p)$, and node separation relatedness $NSR_{A,B}(p)$.

- $HM_{A,B}(p) = \min[\mu_{A,H(A,B)}(p), \mu_{A,H(A,B)}(p)]$. $p$ is a path from node $A$ to node $B$, and $\mu()$ is a membership function.

- $HR_{A,B}(p) = level[H_{A,B}(p)]$. $H_{A,B}(p)$ is the highest level node in the path $p$ connecting nodes $A$ and $B$, and $level[x]$ is the level of node $x$.

- $NSR_{A,B}(p) = \text{Length of the simple path } p \text{ connecting nodes } A \text{ and } B$.

The relatedness is then calculated over each possible path. Finally the interestingness of items $a, b$ in an association rule is defined as:

$$Interestingness(a, b) \propto \frac{1}{Relatedness(a, b)} \tag{2.8}$$

Jaroszewicz and Simovici (2004) presented the use of Bayesian networks for representing DK. Unexpectedness is defined as

$$\mathcal{J}(I, i) = |P^D_I(i) - P^{BN}_I(i)|, \tag{2.9}$$

for the itemset $I$ with values $i$. $P^D_I(i)$ and $P^{BN}_I(i)$ represent the probability in the data and Bayesian network respectively. In definition, the inferred probabilities from Bayesian networks play the role of users’ expectations. Jaroszewicz and Scheffer (2005) leveraged the definition of unexpectedness of events to interestingness of attribute sets: an attribute set $I$ is interesting, if there is an event $I = i$ for which inferred and observed probability diverge:

$$\mathcal{J}(I) = \max_{i \in \text{Dom}(I)} \mathcal{J}(I, i).$$

Abstractly, the divergence of inferred and observed probability in the definitions is identical with the context of Equation 2.1.

Malhas and Aghbari (2009) advanced the BN-based measure of unexpectedness to information theory, i.e. Shannon’s mutual information (Shannon, 1948). By introducing the view of information theory, the context of unexpected patterns is viewed as the increasing of uncertainty to KB when that pattern is entered as new evidence/finding to the Bayesian network of KB. Given a BN, let $Q$ be a query node of the BN and $F$ be an evidence/finding
A sensitivity measure, $S$, is defined for itemsets relative to the BN for gauging the unexpectedness.

$$S(itemsets, values_i) = \sum_{m=1}^{N} \sum_{n=1}^{N} I_{new}(Q_m; F_n)$$

$$= \sum_{m=1}^{N} \sum_{n=1}^{N} H_{new}(Q_m) - H_{new}(Q_m|F_n),$$

where $I_{new}$ is the new mutual information given $F$, and $H_{new}$ is the new entropy.

The log-linear and biased belief models (Xin et al., 2006) approximate the expected frequency $f_e(P)$ of a pattern. Let $f_o(P)$ be the observed frequency of the pattern, the degree of interestingness\(^7\) is calculated by a ranking function $R: R(f_o(P), f_e(P)) \rightarrow \mathbb{R}$; for instance a log-linear ranking function is defined as:

$$R(f_o(P), f_e(P)) = \log f_o(P) - \log f_e(P).$$

### 2.4.3 Knowledge Acquisition and Update

This section highlights the role of a user in mining unexpected patterns. We review previous research in terms of knowledge acquisition and update. Compared to the conventional data-driven data mining paradigm (Figure 2.1(a1,a2)), mining unexpected patterns relies on the existence of KB that comes from a user (Figure 2.1(b)).

In terms of knowledge acquisition, there is a number of well-known problems. One of the problems is that a user (usually an expert) may know a great deal, but it is hard for him/her to relate what he/she knows consistently, let alone to relate everything he/she knows in one setting (Liu et al., 1999a). We can formally decompose this problem into aggregation\(^8\) and consistency\(^9\) subproblems. Another problem is the overhead of knowledge elicitation - that is how much time and effort users are willing to spend on understanding the lan-

---

\(^7\)Notably, the study by Xin et al. (2006) does not actually focus on unexpectedness. It focuses on the patterns that are interesting to the user.

\(^8\)Acquire knowledge iteratively.

\(^9\)Verify the consistency of acquired knowledge.
2.4. REVIEW OF UNEXPECTEDNESS

guage of knowledge representation, and to manually record their knowledge. More detailed survey of knowledge acquisition in addition to the unexpectedness field is presented in Section 2.6. This subsection evaluates previous research of unexpectedness based on aggregation, consistency and overhead problems.

General impressions (GIs) (Liu et al., 1997) are simple and straightforward for users to specify. GIs are designed to improve the method of fuzzy matching (Liu and Hsu, 1996), which is limited by the user’s ability to supply the set of fuzzy expectations. A user is required to write down his/her knowledge about the domain in the format of general impressions, e.g. savings = large \rightarrow\ loan = yes. Then the mined rules can be ranked and matched against the GIs. There is no KB update mechanism in this study.

Liu et al. (1999a) acknowledged that representing and using various types of human knowledge is still a very difficult problem in Artificial Intelligence, and that there is no one representation that can represent all types of knowledge. Therefore, the Interestingness Analysis System (IAS) is proposed. It consists of different tools with different knowledge representations so that a user can express various types of expectations. Liu et al. (1999a) focus on the use of fuzzy patterns as user expectation. Since human knowledge sets or hypotheses are typically imprecise, fuzzy linguistic variables in fuzzy set theory fits naturally. In terms of knowledge acquisition, the user has to write down his/her expectations in the language of fuzzy patterns. Although the proposed IAS is claimed to be capable of iterative knowledge acquisition, this issue is not formally defined in the rest of the paper. Therefore, it is not clear how IAS updates its KB iteratively.

The study of (Liu et al., 2000) integrated various knowledge representations into its Interestingness Analysis System (IAS), such as taxonomy for generalized association rules, general impression (GI), the reasonably precise concept (RPC) and precise knowledge (PK). There is also a visualization system developed for IAS that takes advantage of human visual capabilities for identifying truly interesting rules. (Liu et al., 2000) briefly suggested an approach for knowledge aggregation:

"In each iteration, it (the system) first asks users to specify their existing knowledge about the domain. It then uses this knowledge to analyze the discovered rules according to some interestingness criteria, and through such analysis to identify those potentially
interesting rules.” However, this issue is not further discussed in the rest of the paper.

Objectively, general impressions (Liu et al., 1997) and fuzzy patterns (Liu et al., 1999a) have fewer overheads for users to express their knowledge. In contrast, the study of (Liu et al., 2000) has a high overhead because there are four types of knowledge representations; that means a user needs to learn four types of language for expressing his/her expectation. In response to the aggregation problem, the Interestingness Analysis System (IAS) is designed to work iteratively (Liu et al., 2000). In terms of the consistency problem, GI, RPC, PK and fuzzy patterns do not have mechanisms to check the consistency of acquired knowledge.

Wang et al. (2003) addressed three important issues that have been previously overlooked - namely knowledge dynamics, knowledge push and unexpectedness dynamics. Knowledge dynamic states that whether a rule is unexpected depends on how a user intends to apply the prior knowledge to a given scenario. Knowledge push states that prior knowledge should be considered right from the start of the search to prune uninteresting rules as early as possible. Finally, unexpectedness dynamic declares that the unexpectedness of a rule depends on what other rules have been previously presented to a user. A preference model is proposed to model knowledge dynamics. Since the preference model requires a user to specify the covering knowledge and its covering depth, it surely increases the overhead of knowledge acquisition. In terms of the aggregation problem, a see-and-know assumption is used for masking the remaining unexpected rules which are similar to previously presented rules. This see-and-know assumption is a smart approach because the algorithm can dynamically expend the presented rules as part of the user’s knowledge; it is efficient because users do not have to manually update their expectation after they learn a rule. The consistency problem is not addressed in their study.

Consecutive studies by Silberschatz and Tuzhilin (1995, 1996) promoted the use of the Bayesian approach to knowledge representation which we have briefly described in the previous section. A soft belief needs to be expressed in first-order logic along with a degree of belief: \( P(\alpha|\xi) \) and a weight function \( w \), that influences the importance of the belief. However, the degree of belief and the weight function incur overhead to a user; actually the conditional probability (degree of belief) is difficult to obtain in practice. In terms of the aggregation and consistency problems, an interesting insight is presented in (Silberschatz
and Tuzhilin, 1995); it is discussed in some guidelines for changing the data and belief according to the discovered patterns. When patterns contradict hard beliefs, data needs to be corrected because hard beliefs are unbreakable laws. On the other hand, when patterns contradict soft beliefs, data needs to be checked. The situation is more complicated for modifying beliefs. When patterns contradict soft beliefs, data should be checked first; if data is correct, then update soft beliefs.

Padmanabhan and Tuzhilin (2002) discussed refining knowledge based on unexpected patterns. The refinement strategy is iterative, consisting of three procedures: (a) pattern generation procedure, (b) selection procedure and (c) refinement procedure. The strategy inspects each belief one by one and updates new beliefs accordingly. In the pattern generation step, unexpected patterns are generated for a belief. There are two selection criteria proposed for the selection step. One is the strongest criterion, which takes the rule with the highest confidence and support. The other one is the take-all criterion, which takes all unexpected rules. Finally the selected patterns are appended into the belief in the refinement step. The aggregation problem is resolved in the refinement strategy. The consistency problem is discussed in terms of selection criteria. For example, a new belief is consistent with the original belief if the strongest criterion is applied. Finally, the overhead of knowledge acquisition depends on the complexity of the initial belief, and the overhead of knowledge refinement is low because its refinement strategy is automated.

Jaroszewicz and Simovici (2004); Malhas and Aghbari (2009) used Bayesian networks as knowledge representation. Each node in the network denotes one variable in data. A user can assign links between variables to express his/her belief of associations among variables. The aggregation problem is resolved by the iterative procedure of finding unexpected patterns and manually updating the Bayesian network (some quantitative and qualitative analysis of the update may be applied (Malhas and Aghbari, 2009)). The consistency problem is straightforward because the only critical thing is ensuring the Bayesian network is acyclic. Finally, the overhead problem is relatively small because Bayesian networks can be readily and visually manipulated.

The fuzzy taxonomy (Shekar and Natarajan, 2004) for representing item-relatedness is easy for visualization; therefore, the knowledge elicitation overhead should be low. Although there is no formal design for updating beliefs, there is a brief discussion on the aggregation issue: “...the taxonomy can be
built in an incremental manner and iteratively modified to incorporate additional knowledge.” In terms of the consistency problem, since the fuzzy taxonomy is graphically represented, it is simple for a user to avoid inconsistency unless the taxonomy is too huge.

Xin et al. (2006) designed knowledge acquisition in a different way. KB is learned indirectly from a user’s interactive feedback of discovered patterns. The model presents top k% selected patterns to a user, and the user is required to rank these patterns according to his/her impression of interestingness. Therefore, the knowledge elicitation overhead is proportional to k and the number of iterations which should be reasonably small. This method tackles the aggregation problem by learning the user’s prior knowledge via interactive feedback. Finally, because the KB consists of the log-liner model and biased belief model, which assign a probability to each pattern, there should be no consistency problem.

Summary and Discussion

This subsection examines previous research in the aspects of knowledge acquisition and update. We concentrate on three general problems of knowledge acquisition - namely aggregation, consistency and overhead problems. All previous researchers explicitly or implicitly design their models in an iterative manner, so the user’s prior knowledge can be acquired through time. However, few studies have analyzed or discussed the consistency and overhead issues. We make a comparison table of the reviewed approaches in Table 2.3.

In terms of the automation of knowledge acquisition, previous research could be classified into four types: manually, partial manually, implicit manually, and automated. The manually class is designed so that the entire KB is constructed by users manually (Liu et al., 1997, 1999a, 2000; Silberschatz and Tuzhilin, 1995, 1996; Shekar and Natarajan, 2004). The method in (Jaroszewicz and Simovici, 2004; Malhas and Aghbari, 2009) belongs to the partial manually class because the parameters of a Bayesian network are estimated from data rather than from users. The method of Padmanabhan and Tuzhilin (2002) is also belongs to the partial manually class because after first iteration, the KB will be updated based on previously mined unexpected patterns. Thirdly, the method of Xin et al. (2006) belongs to the implicit manually class because a user does not have to directly elaborate his/her knowledge. Fi-
### 2.4. REVIEW OF UNEXPECTEDNESS

<table>
<thead>
<tr>
<th>KB Representation</th>
<th>Method</th>
<th>Aggregation</th>
<th>Consistency</th>
<th>Overhead</th>
</tr>
</thead>
<tbody>
<tr>
<td>GIs (Liu et al., 1997)</td>
<td>Manually</td>
<td>No</td>
<td>No</td>
<td>Low</td>
</tr>
<tr>
<td>Fuzzy Patterns (Liu et al., 1999a)</td>
<td>Manually</td>
<td>Not Clear</td>
<td>No</td>
<td>Low</td>
</tr>
<tr>
<td>IAS (incl. GI,RPC,PK,etc.) (Liu et al., 2000)</td>
<td>Manually</td>
<td>Yes</td>
<td>No</td>
<td>High</td>
</tr>
<tr>
<td>Knowledge Rules and Preference Model (Wang et al., 2003)</td>
<td>Automated</td>
<td>Yes</td>
<td>No</td>
<td>High</td>
</tr>
<tr>
<td>Hard/Soft Belief (Silberschatz and Tuzhilin, 1995, 1996)</td>
<td>Manually</td>
<td>Yes</td>
<td>Yes</td>
<td>High</td>
</tr>
<tr>
<td>Rule Belief (Padmanabhan and Tuzhilin, 2002)</td>
<td>Partially manually</td>
<td>Yes</td>
<td>Yes</td>
<td>Low</td>
</tr>
<tr>
<td>BN (Jaroszewicz and Simovici, 2004; Malhas and Aghbari, 2009)</td>
<td>Partially manually</td>
<td>Yes</td>
<td>Yes</td>
<td>Low</td>
</tr>
<tr>
<td>Fuzzy Taxonomy (Shekar and Natarajan, 2004)</td>
<td>Manually</td>
<td>Not Clear</td>
<td>Yes</td>
<td>Low</td>
</tr>
<tr>
<td>Log-linear and Biased-belief Models (Xin et al., 2006)</td>
<td>Implicit manually</td>
<td>Yes</td>
<td>Yes</td>
<td>Low</td>
</tr>
</tbody>
</table>

Table 2.3: Comparison table of knowledge acquisition in terms of acquisition method, aggregation, consistency and overhead issues.
nally, the approach for KB update of Wang et al. (2003) could be classified as *automated* because of the *see-and-know* assumption.

### 2.4.4 More Related Work

The study of Dong and Li (1998) defined unexpectedness in terms of neighborhood-based parameters. It is a kind of objective approach which does not take a user’s knowledge into consideration. Sahar (1999) acquired and utilised a user’s expectation in a different way. The prior knowledge is not required; instead, the user is only asked to give some feedback on selected discovered rules. Sahar (1999) asked a user to classify several rules, specifically chosen so their elimination can bring about the automatic elimination of many other rules. The classification task is designed to tell whether: (a) the rule is true, and (b) the user is interested in any rule in the family of the classified rule. Actually, the study of Sahar (1999) is not focusing on unexpectedness; its aim is to find interesting patterns, which is a broader aim than unexpectedness.

### 2.4.5 Comparison of Knowledge Representations

The unexpectedness research that we have reviewed all seems promising and practical. Unfortunately, for data analysts who want to choose among these approaches, there does not exist a standard to compare these state-of-art techniques. Firstly, it is difficult to quantitatively assess an unexpectedness solution by the time and efforts needed for knowledge elicitation because this problem is also subject to the complexity of data and the communication with users.

Secondly, it is hard to compare the usefulness or accuracy of discovered patterns, even when all different solutions have been applied to a same dataset, because the resulting patterns may have different formats, and the evaluation has to be based on the user’s subjective judgment.

Finally, every solution may be designed based on contrasting scenarios, applications, and assumptions, e.g. the user is comfortable in interpreting Bayesian networks (Jaroszewicz and Simovici, 2004) as against the viewpoint that acquiring reasonably precise Bayesian networks is found to be difficult in many real applications (Xin et al., 2006).

This subsection examines existing techniques from the aspect of the utilisation of knowledgebases. Based on the analysis, we choose Bayesian networks as
2.4. REVIEW OF UNEXPECTEDNESS

<table>
<thead>
<tr>
<th>Age</th>
<th>Working</th>
<th>Net Income</th>
<th>Loan</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt; 65</td>
<td>No</td>
<td>$90,000</td>
<td>Yes</td>
</tr>
<tr>
<td>&lt; 65</td>
<td>No</td>
<td>$-30,000</td>
<td>No</td>
</tr>
<tr>
<td>&lt; 65</td>
<td>No</td>
<td>$-50,000</td>
<td>No</td>
</tr>
<tr>
<td>&lt; 65</td>
<td>Yes</td>
<td>$-20,000</td>
<td>No</td>
</tr>
<tr>
<td>&lt; 65</td>
<td>Yes</td>
<td>$15,000</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 2.4: Loan-screening Dataset for comparison of unexpectedness approaches.

the representation because of the probability information that they explicitly store.

We begin with an illustrative example to contrast the capabilities of different representations of earlier work. Suppose there is a five-instance dataset collected for the personal loan screening problem (shown in Table 2.4); Age, Working, Net Income are the factors for evaluation, and Loan is the final result of applications which could be approved or not.

The participants of the KDD project for mining the loan-screening dataset are one loan manager (who is a domain expert and the KDD user) and one data analyst (who operates the KDD system and plays the role of knowledge engineer). A simple, two-phase KDD procedure is applied – phase one acquires the user’s knowledge into a KB, and phase two uses the KB to mine unexpected patterns.

In the first phase, knowledge acquisition, the loan manager provided the data analyst with some domain knowledge, “Customers who are jobless are unlikely to get an unsecured loan. By the way, the elderly, e.g. over 65 years old, are usually retired.” In the hypothetical KDD project, the data analyst would translate the user’s knowledge into different knowledge representations of various approaches.

Using the technique of (Liu et al., 2000), the data analyst translated the DK into RPC (reasonably precise concept):

\[
\text{rpc}(\text{Working} = \text{No} \rightarrow \text{Loan} = \text{No})[\text{support} = ?, \text{confidence} = ?]
\]

\[
\text{rpc}(\text{Age} > 65 \rightarrow \text{Working} = \text{No})[\text{support} = ?, \text{confidence} = ?]
\]

Similarly, the DK could be translated into knowledge rules or ‘beliefs’ (Wang et al., 2003; Padmanabhan and Tuzhilin, 1999) in the format of association
rules:

\[ \text{Working} = \text{No} \implies \text{Loan} = \text{No} \quad (R_1) \]
\[ \text{Age} > 65 \implies \text{Working} = \text{No} \quad (R_2) \]

Using the technique of (Jaroszewicz and Simovici, 2004), the data analyst constructed a BN with the edges: \text{Working} linked to \text{Loan} and \text{Age} linked to \text{Working} (shown in Figure 2.6) where the parameters of the BN are to be learned from the data. In terms of taxonomy representation (Shekar and Natarajan, 2004), it is not clear how to represent the DK in a taxonomy.

After the first phase of knowledge acquisition, suppose conventional association rule mining was applied to the dataset, and the data analyst wanted to gauge the unexpectedness of a particular mined rule:

\[ \text{Age} > 65 \implies \text{Loan} = \text{Yes} \quad [\text{support} = 20\%, \text{confidence} = 100\%]. \]

We will call it the elder-approved rule in the following discussion.

By the definitions of (Liu et al., 2000; Wang et al., 2003)\textsuperscript{10}, the elder-approved rule has to be compared to every piece of knowledge in the KB respectively. The comparison checks two parts – the body (condition) part and the head (consequent) part of the rules. Let \( R_i \) be the \( i \)th rule in the KB, if the elder-approved rule matches the body part of \( R_i \) but not the head part (or vice versa), then the rule should be termed unexpected with respect to \( R_i \).

The table below shows the comparison between the elder-approved rule and the first piece of RPC/‘knowledge rule’ in the KB.

\textsuperscript{10}We outline the fundamental mechanisms of earlier work in this discussion without mentioning its details.
Since the body and head parts are both not matching, the elder-approved rule is not unexpected with respect to \( R_1 \) of the KB.

To the contrary, the elder-approved rule is unexpected with respect to \( R_2 \) of the KB; because the body part matches but the head part does not.

However, the reason above for deciding the elder-approved rule is unexpected seems counterintuitive because \textit{Working} and \textit{Loan} are two distinct concepts.

According to the approach of Padmanabhan and Tuzhilin (1999), \( R_1 \) should be taken into computation. Whether the elder-approved rule is unexpected is subject to a user-specified threshold used in condition (2) (see the table below); for example, if the threshold is set to 25\%, then the rule is not unexpected.

<table>
<thead>
<tr>
<th>( R_1 ) rule</th>
<th>Working(=)No (\rightarrow) Loan(=)No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age(&gt;)65 (\rightarrow) Loan(=)Yes</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( R_2 ) rule</th>
<th>Body (condition)</th>
<th>Head (consequent)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( R_2 ) (\in) KB</td>
<td>Age(&gt;)65</td>
<td>Working(=)No</td>
</tr>
<tr>
<td>elder-approved rule</td>
<td>Age(&gt;)65</td>
<td>Loan(=)Yes</td>
</tr>
<tr>
<td>matching</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

Obviously, the earlier work of rule based representations designed the calculation of unexpectedness \textit{locally}. That is, an incoming rule is compared to each piece of knowledge in the KB separately, and as long as the incoming rule is unexpected to at least one existing rule of the KB, it is termed unexpected. However, utilising the KB in this way does not capture the feature of how humans reason. One would possibly infer the conclusion that the elderly would be unlikely to get an unsecured-loan because they are no longer working, from the DK provided by the expert.

Oppositely, Bayesian network based representation would use the two pieces of knowledge, \( R_1 \) and \( R_2 \), together to reason about the elder-approved rule.
In computation, the data analyst could set the \texttt{Age} node as the evidence node and then query the \texttt{Loan} node. The knowledge of the \texttt{Age} node, $> 65$, will propagate through the \texttt{Working} node, then reach the \texttt{Loan} node returning an inferred conditional probability. Computation of unexpectedness could be done by contrasting the confidence of the rule and the inferred conditional probability\textsuperscript{11}. Using the chain rule theorem provided by (Korb and Nicholson, 2003, chap. 1), the inference of conditional probability in the BN (Figure 2.6) is calculated:

$$
\hat{P}(\text{Loan} = \text{Yes}|\text{Age} > 65) = P(\text{Loan} = \text{Yes}|\text{Working} = \text{Yes})P(\text{Working} = \text{Yes}|\text{Age} > 65)
+ P(\text{Loan} = \text{Yes}|\text{Working} = \text{No})P(\text{Working} = \text{No}|\text{Age} > 65)
= 1 \cdot 0 + 0.25 \cdot 1 = 0.25
$$

Therefore, the conditional probability inferred from the BN suggests there is only 25% of a chance that age $> 65$ elders would get an unsecured-loan. In this situation, the data analyst would conclude that the elder-approved rule is unexpected with respect to the inferred probability.

One might question why the data analyst should not apply forward chaining in logic (Black, 1987) to the task of unexpectedness. For example,

\begin{itemize}
  \item \texttt{Age}$> 65$ implies Not working
  \item Not working implies Loan disapproved
  \item therefore, \texttt{Age}$> 65$ implies Loan disapproved.
\end{itemize}

Contrasting to the conclusion above, with forward chaining, the elder-approved rule would be unexpected. Indeed, using logic would resolve the ‘local’ reasoning problem. However, the rules in the KB are not sentences in logic; instead, they have the context of association rules – that is, these rules are not 100% valid all the time.

Transitivity, shown in the example above, is a property of forward chaining. However, generic rules which are not 100% valid do not have the transitivity property. For example, suppose the rule $A \rightarrow B$ has confidence 0.8; it means

\textsuperscript{11}The described method of unexpectedness for association rules is different from the work of Jaroszewicz and Simovici (2004) which deals with frequent itemsets.
2.4. REVIEW OF UNEXPECTEDEDNESS

\[ P(B|A) = 0.8. \] If generic rules were transitive, we should be able to infer \( A \rightarrow C \): \( P(C|A) > \theta \) when we know \( A \rightarrow B \) and \( B \rightarrow C \) where \( \theta \) is the threshold for confidence; however, it is not so. Figure 2.7 demonstrates why generic rules do not have the transitive property. Both Figures 2.7(a) and 2.7(b) are cases where the confidence of the two rules \( A \rightarrow B \) and \( B \rightarrow C \) are above the threshold \( \theta \) but \( A \rightarrow C \) is not. Therefore, without transitivity, rule-based representations (used in the unexpectedness literature) could only calculate unexpectedness by comparing the query rule against every rule of KB one by one.

![Diagram showing transitivity](image)

Figure 2.7: Diagrams showing that generic rules do not have transitivity property. Two possible cases for ARs \( A \rightarrow B \) and \( B \rightarrow C \); a cycle denotes a set.

An insight into the reason for why BNs are capable of making inference transitivity but generic rules are not, is that BNs encode more information than generic rules do. That is the conditional probabilities of all values between linked variables are explicitly stored in the BN. In comparison, generic rules do not possess the conditional probabilities of all values for a pairwise relation. For example, the representation of rule \( A \rightarrow B \) may optionally possess the value of \( P(B|A) \), but other combinations of conditional probabilities, e.g. \( P(\neg B|A), P(B|\neg A), P(\neg B|\neg A) \), are not recorded. In conclusion, because BNs encode pairwise knowledge via edges/links and joint probabilities, which enables transitivity, via its CPTs, we choose them as the representation in our research of unexpectedness.

\[ ^{12} \text{E.g. } A, B \text{ are associated.} \]
2.4.6 Principles for Unexpectedness

Based on the review and analysis of previous research about unexpectedness, this subsection summarizes the general principles for designing a system for mining unexpected patterns. We also present our point of view for designing a knowledge-rich and user-interactive data mining paradigm.

In previous research, there are different scenarios for knowledge discovery, e.g. classification rules versus frequent itemsets, and various procedures with different knowledge representations; the design of knowledge acquisition and update varies as well. In order to generalise the principles for unexpectedness, we start by discussing its fundamental issues, which are listed in the following.

Interface Overhead - The interface overhead exists in between a user and the data mining system. The first overhead exists in the direction from users to KB. For example, a user needs to learn the language of a particular knowledge representation, and spends time in writing down his/her knowledge; this procedure usually needs assistance from data analysts or knowledge engineers. The second overhead occurs in the interpretation of unexpected patterns by users. A user may not be familiar with the language or the meaning of patterns; again, data analysts have to assist pattern interpretation. In sum, we state the first principle: a data mining system should be designed based on the context of usability, i.e. easy to learn and effortless.

Knowledge Dynamics - There are three elements that change over time in mining unexpected patterns - namely DK, KB and mined rules. Initially, the KB has to be acquired somehow. Second, a mining algorithm calculates the degree of unexpectedness for the mined patterns against the KB and presents results to a user. Third, the user learns some knowledge from the unexpected patterns. Finally, the user refines the KB and performs the next iteration for discovering new unexpected patterns. Therefore, the user’s knowledge evolves dynamically, as well as the KB and unexpected patterns. Accordingly, we address the second

---

13We use this phrase with a slightly different meaning with Wang et al. (2003).

14This statement assumes that the data is static.
principle as: **a data mining system should be capable of handling the knowledge dynamics.**

**Representation Capabilities** - We have discussed that being able to use every piece of knowledge collectively is a more powerful way of using the KB (Subsection 2.4.5); transitivity was recognised as a necessary condition. However, not all representations are capable of reasoning transitive relations; that means some knowledge representations could only use their knowledge *locally*. Toward knowledge-rich data mining, it is desired to have a reusable KB and a data mining system that can use the KB intelligently. Therefore, the third principle would be: **a data mining system should utilise knowledge representations with fewer limitations and higher capabilities.**

Domingos (2007) has explained that the goal of knowledge-rich data mining is to support a feedback loop by which a small amount of initial knowledge can be bootstrapped into more knowledge by mining. Coincidentally, mining unexpected patterns has been researched in the context of knowledge-rich data mining where unexpectedness originates from post-processing. Although there is no unified research that addresses all different aspects of unexpectedness, most existing research has more or less tackled some important issues. The current achievements of unexpectedness all assume that data is static and the solution is for a single standalone data mining project\textsuperscript{15}. Nevertheless, a more realistic scenario is a dynamic database environment (Kriegel et al., 2007); thus, keeping patterns up-to-date and finding ‘patterns of evolving patterns’ are important challenges for the future. Acknowledging the challenges of future research, however, the thesis sticks to the scope of static data.

**2.4.7 Summary**

Mining unexpected patterns is subject to the knowledge and sometimes the interest of a user. Therefore, it is closely related to the problems of knowledge representation, knowledge acquisition, knowledge aggregation/refinement and

\textsuperscript{15}There is an exception that Silberschatz and Tuzhilin (1996) have extended their research to a data monitoring and discovery triggering system.
user-interaction. The knowledge of a user changes dynamically because of new patterns; thus, the data mining system has to adapt to new knowledge from the user and provide a new set of unexpected patterns accordingly.

We have generalized the research of unexpectedness to three major issues: (a) interface overhead, (b) knowledge dynamics and (c) representation capabilities. Accordingly, a system for mining unexpected patterns should utilize a powerful knowledge representation that can provide efficient human-computer interaction and adapt to a user’s new knowledge effortlessly.

In Chapter 5, we further develop the approach based on Bayesian networks, and propose improvements in both theoretical and pragmatic aspects for mining the real-world datasets we have explored.

2.5 Explanations for Association Rules

Finding explanations for association rules has not received much attention in the data mining community; however, explaining a new discovery is a central issue in the practice of science. In this section, we report two pioneering works toward this aim.

Firstly, Yao et al. (2003) propose using a reverse process of conditional association for finding explanations for association rules. They define an explanation as a condition $\chi$ that slices the data cube so that the support $s(\phi|\psi)$ of an association rule $\phi \rightarrow \psi$ becomes higher: $s(\phi|\psi|\chi) > s(\phi|\psi)$.

Example 1. In our test of the method of (Yao et al., 2003), we applied it to WWW User Survey data, which is introduced in the next chapter. There is a rule: User is not willing to pay because of free sources $\rightarrow$ Male (support=0.101, confidence=0.637). The top 6 explanations are listed in Table 2.5; they are about variables Job and Country where these conditions can increase the support of conditional association, $s(\phi|\psi|\chi) > s(\phi|\psi)$.

Notably, these explanations only cover 19 instances of the rule, while there are still 2658 instances not covered by these explanations.

The objective of data mining is in fact the goal of scientists when carrying out scientific research, independent of their various disciplines. Data mining, combining research methods and computer technology, by its nature, should be considered as a research support system (Yao and Zhao, 2005). Therefore,
Table 2.5: Explanations found by the method of Yao et al. (2003).

| Condition | $s(\phi_1 | \chi)$ | Covered instances |
|-----------|-------------------|-------------------|
| Job=Other $\land$ Country=South Korea | 1 | 1 |
| Job=Other $\land$ Country=Dominican Republic | 1 | 1 |
| Job=Other $\land$ Country=Egypt | 1 | 1 |
| Job=Other $\land$ Country=Greece | 1 | 1 |
| Job=Other $\land$ Country=Sweden | 0.667 | 12 |
| Job=Other $\land$ Country=Portugal | 0.667 | 3 |

Yao et al. (2003); Yao and Zhao (2005) propose an explanation-oriented data mining framework (depicted in Figure 2.8) by the method described above.

Figure 2.8: Explanation-oriented data mining framework (Yao and Zhao, 2005).

Another approach applies a domain ontology to serve as explanations for association rules (Svátek et al., 2005a). The method is to pull a set of explanation template from an ontology base for a rule. For example, Figure 2.9 shows a set of explanation template for the rule Bad Living Standard Expectancy $\rightarrow$ Dislike KSCM Party. The demonstrated approach is processed manually; therefore, it requires much effort to map the ontology to data, extracting
CHAPTER 2. KNOWLEDGE RICH DATA MINING

explanation templates, and interpreting rules based on these templates.

<table>
<thead>
<tr>
<th>Template</th>
<th>Mapped</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>KSCM ∈ Political.party hasPartyProgramme</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>Party_programme □ Plan_of.action hasObjective</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Social.phenomenon ⊃ BAD.LIVING.STANDARD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSCM ∈ Political.party isRepresentedIn</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>Administrative.body □ City.council carriesOutAction</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Economic.action hasImpactOn Social.phenomenon ⊃ BAD.LIVING.STANDARD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSCM ∈ Political.party hasPartyProgramme</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>Party_programme □ Plan_of.action envisagesAction</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Action □ Economic.action hasImpactOn Social.phenomenon ⊃ BAD.LIVING.STANDARD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSCM ∈ Group informsAbout Social.phenomenon ⊃ BAD.LIVING.STANDARD</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>KSCM ∈ Group carriesOut Action □ Economic.action hasImpactOn Social.phenomenon ⊃ BAD.LIVING.STANDARD</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>KSCM ∈ Group participatesIn Event □ Economic.action hasImpactOn Social.phenomenon ⊃ BAD.LIVING.STANDARD</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>KSCM ∈ Group supports Action □ Economic.action hasImpactOn Social.phenomenon ⊃ BAD.LIVING.STANDARD</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>KSCM ∈ Group fightsAgainst Group carriesOutAction</td>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>Action □ Economic.action hasImpactOn Social.phenomenon ⊃ BAD.LIVING.STANDARD</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 2.9: Explanation templates for rule Bad Living Standard Expectancy → Dislike KSCM Party.

2.5.1 Summary

To the best of our knowledge, there are two approaches in previous work to explanation generation for association rules. One is based on the conditioning of attributes to increase the conditioned support of a rule (Yao et al., 2003); another approach uses domain ontologies for providing templates of explanations (Svátek et al., 2005a).

Unlike the approaches of previous research, we present a novel idea of generating explanations in the format of reasoning chains that provide relevant factors via probabilistic models (in Chapter 6).

2.6 Knowledge Acquisition

This section briefly reviews some recent development in knowledge acquisition. Knowledge representation is the first issue examined in the review; broadly
speaking, there are two streams of representations: computable knowledgebases and natural language.

The process of eliciting knowledge from experts for building knowledgebases is called knowledge acquisition (Russell and Norvig, 1995, chap. 8). From the earlier development of expert systems to recent thriving research in organisational knowledge management, knowledge acquisition plays a critical role; however, knowledge acquisition is usually a bottleneck in building up such a knowledgebase. All the same, this issue surely exists in the research of unexpectedness; therefore, this section reviews some recent research about knowledge acquisition from the perspective of its background, the present trends, and its obstacles.

In terms of knowledge representation, mainstream research acquires experts’ knowledge into formal representations based on traditional artificial intelligence, e.g. ontologies, the decision tree, fuzzy logic, the condition-action (IF-THEN) rule, Prolog (Richards, 2004; Xing et al., 2003; Yao et al., 2005; Baral et al., 2007). On the other hand, natural language is often used as a primitive representation in the domain of knowledge management. There is a new trend that captures organisational knowledge in the format of natural language via collaborative and conversational knowledge management, e.g. Wiki and Blogs (Wagner, 2006; Hasan and Pfaff, 2006; IP and Wagner, 2008; Bhagdev et al., 2007). The school of formal knowledge representations uses knowledgebases for autonomous and intelligent computations for decision support systems, supervisory control systems and many other applications. On the other hand, the new approach with natural language stresses the virtue of capturing and maintaining exponentially growing volumes of knowledge into document-based knowledgebases through collaboration and conversation.

A knowledge acquisition methodology is subject to some key features of its application; they are (a) the characteristic of knowledge of interest, (b) the intended utility of the knowledgebase, and (c) the preferred communication protocol. In general, only a small portion of human knowledge could be acquired due to the limitation of the knowledge representations and the narrow bandwidth of communication channels. For example, Ishino and Jin (2002) studied the types of engineering knowledge from the dimension of application functions. There are two large-scaled categories in engineering design: domain knowledge and strategic knowledge; under the two categories, ten subcategories of knowledge are further identified (shown in Figure 2.10). Among
all the identified categories of knowledge, Ishino and Jin (2002) only focused on capturing the *know-why* knowledge. From this example, we could realise that knowledge acquisition has to focus on the manageable selection of types of knowledge because the resources are limited; and the design of knowledge acquisition is consequently predetermined by the characteristic of the selected knowledge type.

![Figure 2.10: The categories of knowledge in engineering design (Ishino and Jin, 2002).](image)

<table>
<thead>
<tr>
<th>Large-Scaled Category</th>
<th>Medium-Scaled Category</th>
<th>Small-Scaled Category</th>
<th>Example No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain Knowledge</td>
<td>Design Principle Knowledge</td>
<td>Function Knowledge</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Product Knowledge</td>
<td>Constraint Knowledge</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Process Knowledge</td>
<td>Accompanying Knowledge</td>
<td>3</td>
</tr>
<tr>
<td>Know-how Knowledge</td>
<td>Task Decomposition Knowledge</td>
<td>Dependency Knowledge</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Know-why Knowledge (Rationale)</td>
<td>Design procedure</td>
<td>5</td>
</tr>
<tr>
<td>Strategic Knowledge</td>
<td>Knowledge of Procedure Features</td>
<td>Knowledge for Reasons behind Procedure</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Knowledge for Reasons behind Products</td>
<td>Knowledge for Reasons behind Products</td>
<td>7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example No.</th>
<th>Example of Knowledge</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Layout the major functions first, and then get into auxiliary functions.</td>
</tr>
<tr>
<td>2</td>
<td>If volume, ( V ), is fixed, pressure, ( P ), is calculated by the equation, ( P = \frac{nRT}{V} ).</td>
</tr>
<tr>
<td>3</td>
<td>The required reduction ratio should be accomplished strictly within ( \pm 4.0% ).</td>
</tr>
<tr>
<td>4</td>
<td>Though the shape of a container doesn’t affect the main function, a streamlined form seems preferable.</td>
</tr>
<tr>
<td>5</td>
<td>For decomposing gear design task, Plan P will cause less dependencies than Plan 5 does.</td>
</tr>
<tr>
<td>6</td>
<td>Activity A, deciding a gear material, and Activity B, determining a pitch of the gear, are mutually dependent.</td>
</tr>
<tr>
<td>7</td>
<td>Activity C, examining the effect of gear teeth, should be tried before Activity D, examining the effect of a gear material.</td>
</tr>
<tr>
<td>8</td>
<td>In order to reduce the cost, it is effective to change the gear material from No.1 to No.2.</td>
</tr>
<tr>
<td>9</td>
<td>Once Angle E had been designed as 60 degrees and then it was changed into 45 degrees. It was because at first the former condition seemed to provide comfortable space for users, and then the designer realized that 45 degrees was sufficient and it could work more compatibly with other parts.</td>
</tr>
<tr>
<td>10</td>
<td>Although lighter, smaller and cheaper is better, the most important factor the cost.</td>
</tr>
</tbody>
</table>

The intended utility of knowledgebases, another factor for knowledge acquisition, predetermines the selection of knowledge representation; then, the communication protocol is developed according to the selected representation. Ishino and Jin (2002) proposed a *three-layer design process model* for the application of computer-aided design (CAD) in the engineering domain. Based on the goal of not disrupting a normal design process, Ishino and Jin (2002) de-
vised a design called GEDP (Grammar and Extended Dynamic Programming), which acquires know-how knowledge from the event log of a CAD application. Also, in the manufacturing domain, Xing et al. (2003) presented an integrated knowledge acquisition approach for building up a knowledge-based system for training workers’ skill in solving the problems in drop hammer forming. The approach of (Xing et al., 2003) combines the traditional interview-based knowledge acquisition with data mining that generates rules for fuzzy inference. Figure 2.11 shows the proposed system that can automatically acquire knowledge from raw data, and can have its rules interactively revised by domain experts. In the design of (Xing et al., 2003), the communication protocol with domain experts follows the traditional interview-based approach that a knowledge engineer interviews domain experts, and then transfers the acquired unstructured knowledge into structured knowledge in the IF-THEN format.

Differently from the concept of computable knowledge, Hasan and Pfaff (2006) summarised that the emerging trends in organisational knowledge management are to create a sense-making model of collective knowledge creation, disruption and utilisation that allows a pragmatic and conceptual alternative to scientific management. Therefore, the Wiki (Hawaiian word for ‘fast’) technology (Leuf and Cunningham, 2001) is proposed to be a new platform for knowledge acquisition and management in organisations (Wagner, 2006; Hasan and Pfaff, 2006). Compared with traditional knowledge management systems, Wiki places less emphasis upon centralised control, strict discipline, and extensive monitoring of the systems to manage knowledge in the organisation.

2.6.1 Knowledge Acquisition Bottleneck

Here we review some recent research that addresses the problem of knowledge acquisition bottleneck. Wagner (2006) summarised four points of knowledge acquisition bottleneck in organisational knowledge management:

**Narrow bandwidth.** The channels that exist to convert organizational knowledge from its source (either experts, documents, or transactions) are relatively narrow.

**Acquisition latency.** The slow speed of acquisition frequently is accompanied by a delay between the time when knowledge
Figure 2.11: The integrated knowledge acquisition system in Xing et al. (2003).

(or the underlying data) is created and when the acquired knowledge becomes available to be shared.

**Knowledge inaccuracy.** Experts make mistakes and so do data mining technologies (finding spurious relationships). Furthermore, maintenance can introduce inaccuracies or inconsistencies into previously correct knowledgebases.

**Maintenance trap.** As the knowledge in the knowledgebase grows, so does the requirement for maintenance. Further, previous updates that were made with insufficient care and foresight will accumulate and render future maintenance increasingly more difficult.

To overcome the bottleneck, a domain ontology is used as a starting point (Suraweera et al., 2004; Boicu et al., 2001). In the application of Intelligent Tutoring Systems (ITS), Suraweera et al. (2004) proposed the use of a domain
ontology to develop the knowledgebase which is based on constraint based modeling (CBM) (Greer and McCalla, 1994). In the design, the first step is to manually compose the domain ontology by domain experts; then the ontology is used to assist the production of constraints with a graphical editor. It is concluded that ontologies enable experts to visualise the constraint set and to reflect on the domain, assisting them to create more complete constraint bases. Boicu et al. (2001) also presented an approach to the rapid development of knowledge-based agents by experts that is based on ontology reuse and development. The agent development platform, Disciple-RKF, contains a general problem solving engine, a learning engine, and an initially empty knowledgebase. The platform is designed for experts to teach agents how to solve problems described in an ontology. The development of the ontology is via the platform’s browsers and viewers for easy navigation, visualization and modification by experts.

The previous related work of ontologies, called ‘ontological engineering’, emerges as a solution to the knowledge acquisition bottleneck (Richards, 2004). In that solution, the ontology is first developed, then used to guide the knowledge acquisition process. However, ontologies are time-consuming to develop, difficult to specify and verify, and harder to validate and maintain (Richards, 2004). Therefore, a new problem, termed as ontology acquisition bottleneck, is addressed; and Richards (2004) proposed to resolve it by reverse ontological engineering. The proposed reverse engineering approach automatically generates an ontology base from rule bases; the rule bases are built up by ripple-down rules (RDR) (Compton et al., 1992), which are a rapid and incremental knowledge acquisition and representation technique.

Automatic knowledge acquisition is another common strategy to overcome the knowledge acquisition bottleneck. Data mining and machine learning could be integrated with manual acquisition (Xing et al., 2003). In the design of (Xing et al., 2003), rules are extracted from trained neural networks by their proposed algorithm; alternatively, rules can be generated by decision tree learning. Towards the goal of automatically extracting knowledge from natural language text, Baral et al. (2007) investigated how to automatically generate Answer Set Prolog (AnsProlog) by the CCG parser (Clark and Curran, 2007) and the semantic analysis tool – Boxer (Bos et al., 2004). This design works in three steps: (1) the CCG parse, together with taggers and other tools, is used to translate English text into basic logic form; then (2) the Boxer tool
parses the basic logic form into first order logic representation; finally (3) the first order logic representation is translated into a set of AnsProlog facts.

Collaboration has been proposed as a new approach to tackle the knowledge acquisition bottleneck. Recent success in open-source collaboration, e.g. the Wikipedia, has drawn some attention from the field of knowledge management (Wagner, 2006; Hasan and Pfaff, 2006). (Wagner, 2006) argued that artificial intelligence and expert systems cannot overcome the challenge of capturing an organisation’s knowledge on a large scale and making it available to the entire organisation. Therefore, the Wiki approach is researched as a suitable technology for collaborative knowledge management in a way that many people work together to create or acquire knowledge instead of a few individual experts. Hasan and Pfaff (2006) reported evidences showing that many companies have successfully used Wikis to work collaboratively and shown how the Wiki will ‘write itself’, depend on its users to contribute and maintain this growing repository of knowledge in the organisation. However, Hasan and Pfaff (2006) also reported a counterexample of its own project where the management of an organisation opposed the trial of setting up a Wiki for the following reasons. The management concerns were (a) limit power sharing - the use of a Wiki may flatten the organisational hierarchy, changing traditional and hierarchical communication channels, and (b) centralised information control - the organisation maintains that it offers better quality control in its existing approach to documentation management with formal editing opportunities, review and verification stages.

2.6.2 Summary

The designs of knowledge acquisition could be very different due to three factors: (a) the characteristic of knowledge of interest, (b) the intended utility of the knowledgebase, and (c) the preferred communication protocol. Generally, the design of a knowledge acquisition system usually concentrates on a specific kind of knowledge: one particular application, e.g. decision support or automatic control; and one preferred communication protocol. If the target knowledge is versatile, natural language would be a solution, and it has been researched in the domain of knowledge management.

Aiming to overcome knowledge acquisition bottleneck, three approaches have been researched. The first approach uses intermediate knowledgebase, i.e.
ontology bases; this method first develops an ontology base then uses this base to guide the knowledge acquisition process. The second strategy is automatic knowledge acquisition; this approach utilises data mining or machine learning to generate knowledge from data. Finally, the collaboration of a group of experts is also a method to resolve the bottleneck, such as the Wikipedia.

In terms of the design of unexpectedness, the user would need to input initial knowledge to kick-start the system for getting initial unexpected patterns. The trade-off is between the amount of knowledge encoded and the quality of the mined rules. In the research, the objective of the design of knowledge acquisition is to minimise the overhead during communication. In addition, here we point out an interesting observation that the nature of the iterative design of knowledge update in unexpectedness is itself a solution to address the problem of knowledge acquisition bottleneck: because patterns might provoke the user’s memory of knowledge for updating the knowledgebase, then the updated knowledgebase could be applied to find different patterns. We propose a user-friendly design of knowledge acquisition for unexpectedness in Chapter 5.

2.7 Ontology in Data Mining

Domain ontology is a hot topic in today’s knowledge engineering research because it is capable of expressing domain concepts and relationships in a way that is consensual and comprehensible to the given professional community. It enables a machine to use the knowledge and enables multiple machines to share their knowledge. With the availability of large scale and shared ontologies, much attention has been drawn to how to apply an ontology in the process of knowledge discovery.

2.7.1 Domain Ontology

Kudenko (2000) applied an ontology for feature generation in constructive induction. Domain expertise is first captured in an ontology using common knowledge acquisition techniques. A subset of this ontology that is suitable for the respective machine learning task is then selected semi-automatically by a domain expert. The reduced ontology is used as a basis for feature generation. Each generated feature corresponds to a concept created from the ontology.
Phillips and Buchanan (2001) tried to reduce the amount of time required of a domain expert by starting with data in a database and inferring facts and relations about the variables using an underlying ontology. The goals are: (a) to automatically suggest and generate new attributes based upon semantic and domain information, (b) to capture useful knowledge for reuse, and (c) to reduce the user’s workload in interpreting new tables. The system scans new databases to obtain type and constraint information then uses this information in the context of a shared ontology to intelligently guide the potentially combinatorial process of feature construction.

Another usage of ontologies is studied in (Cespivova et al., 2004). It studied the use of a medical ontology and other background knowledge in the process of association rules mining. The UMLS (Unified Medical Language System), which provides terminological taxonomies\(^\text{16}\), is used to semantically group-up variables. The grouped variables then can decompose a general mining task into more specific tasks – for example, mining the association between ‘Activity’ group and ‘Disease or Syndrome’ group. Presumably, the mined hypotheses entering the evaluation phase will be smaller and more homogeneous, hence easier to examine for a human evaluator. A following study by (Svátek et al., 2005b) further applied the design to the social climate application. The issue of ‘result deployment over semantic web’ is also discussed in two options. First, a promising approach would be to incorporate the mining results into semantic web documents. The most straightforward way to do so is to take advantage of analytic reports – textual documents presenting the results of the KDD process in a condensed form. Secondly, it is possible to use association rule mining to learn (skeletons of) OWL\(^\text{17}\) ontologies from data. The knowledge contained in the analytic reports would then be represented as ontology axioms rather than as rules, which would enable us to exploit description logic reasoners to formally compare the sets of results.

Brisson and Collard (2008) presented a methodology for integrating expert


\(^\text{17}\)http://www.w3.org/2004/OWL/
knowledge all along the data mining process in a coherent and uniform manner. Within the methodology, an ontology driven information system (ODIS) is proposed based on an application ontology, a knowledgebase and a mining oriented database rebuilt from source raw data. Thus, experts’ knowledge is used during business and data understanding, data preparation and result evaluation steps. In the business understanding step, knowledge is formalised in the format of production rules, also called IF-THEN rules. These rules are modular, and define a small and independent piece of knowledge, assigned with a confidence level and certainty level for the evaluation of interestingness. In the data understanding step, an ontology is built to identify domain concepts and relationships; the objectives are to select interesting attributes according to the business objectives, to solve ambiguities within data and to choose data discretisation levels. A mining oriented database (MODB) is also built to structure knowledge and data in order to process efficient mining tasks and to save time spent in data preparation. Finally, in the result evaluation step, the mined rules are compared against the knowledgebase.

2.7.2 Data Mining Ontology

Aiming to assist novices and data mining specialists in the steps of choosing possible combinations of data preprocessing, mining algorithms and postprocessing methods, Bernstein et al. (2005) presented an ontology-based intelligent discovery assistant (IDA). IDA determines the characteristics of the data and of the desired mining result, and uses the ontology to search for and enumerate the data mining processes that are valid for producing the desired result from the given data. Therefore, IDA can generate valid, non-trivial, and sometimes interesting DM-process instances.

2.7.3 Summary

To summarise, ontologies have been researched for feature generation, variable grouping, result deployment, data understanding and DM-process synthesis. In addition, some research applies domain ontologies to explain association rules; we leave the discussion to Subsection 2.5. In Chapter 4, we research the ontology for addressing the meaningfulness problem of mined rules via the variable grouping technique.
2.8 Chapter Summary

In this chapter, the essential background of the thesis is presented. We broadly discuss the role of knowledge in KDD. Subjective interestingness is briefly introduced in Section 2.2 and we propose a nomenclature (Subsection 2.2.1) to distinguish the terms related to ‘surprising’/‘unexpected’ throughout the text of the thesis. We then look into the topics – surprising pattern mining (data-driven and unexpectedness), explanation generation, knowledge acquisition, and ontology in data mining from Sections 2.3 to 2.7.

The survey of ontologies in DM (Section 2.7) connects to our study of ontology-driven DM in Chapter 4. The review of unexpected pattern mining (Section 2.4) and knowledge acquisition (Section 2.6) is related to the chapter of designing unexpected pattern mining (Chapter 5). Section 2.5 of explanation generation links to Chapter 6 which presents a new approach to explain mined rules. Before presenting our work in mining surprising patterns, we introduce the datasets and methods which will be used in the case studies in the next chapter (Chapter 3).
Chapter 3

Datasets

This chapter introduces the datasets that are used in the various case study analyses throughout this thesis. There are three categories of case study analysis described: comparison, demonstration, and evaluation. ‘Comparison’ refers to studies which compare the proposed data mining methods against related work; ‘demonstration’ refers to studies that demonstrate the work flow of proposed unexpectedness and explanation methods; and ‘evaluation’ refers to the evaluation of knowledge discovery designs based on the participation of domain experts.

Domain experts play a major role in the proposed KDD design for mining surprising patterns, and therefore the majority of the analysis focusses on two real-world datasets from domains in which experts were available and willing to be involved: the first a large clinical database related to treatment of kidney disease (supported by a clinical nephrologist), the second a comprehensive study of how mathematics students understand decimal notation (supported by two educational experts). Additionally, three datasets for which no domain expertise was available were used: one related to geriatric health for comparison against existing unexpectedness studies; and two others related to population census data and world wide web usage respectively, for demonstration of specific points in mining surprising patterns.

3.1 Materials

We use five datasets in the thesis.

ANZDATA. The Australia and New Zealand Dialysis and Transplant Reg-
Anzdata (ANZDATA\textsuperscript{1}) is collected by survey form for each dialysis and transplant patient at yearly intervals. The dataset used in the thesis is a snapshot of ANZDATA, and contains 217,083 records of 19,220 kidney disease patients spanning 12.6 years; it has 217,083 entries (rows) and 96 attributes (columns).

**DCT.** During 1995 to 1997, the Decimal Comparison Test (DCT) was used to gather data on students in Victoria (of Australia) from Grades 5 to 10 in order to understand how students learn decimals. In the test, students are asked to choose the larger number from each of 30 pairs of decimals. The pairs of decimals are carefully chosen so that from the patterns of responses, students’ (mis)understanding can be diagnosed as belonging to one of a number of categories (Steinle and Stacey, 1998). The dataset has 3,531 entries (rows) and 40 attributes (columns).

**KSL.** The KSL dataset comes from a study measuring health and social characteristics of representative samples of Danish geriatrics, taken in 1967 and 1984; it is the same dataset used by Jaroszewicz and Simovici (2004). The dataset has 1,083 entries (rows) and 9 attributes (columns).

**Census Income.** The Census Income (or called Adult) dataset contains weighted census data extracted from the 1994 and 1995 Current Population Surveys conducted by the U.S. Census Bureau; it is downloadable from the UCI Machine Learning Repository\textsuperscript{2}. The data contains 14 demographic and employment related variables; the dataset has 32,562 entries (rows) and 14 attributes (columns).

**WWW User Survey.** The WWW User Survey data comes from Graphic, Visualization, and Usability Center’s (GVU) 8th WWW User Survey\textsuperscript{3} that collects the web users’ background, interest on internet shopping and related information; The Survey was run from October 10, 1997 through

\textsuperscript{1}http://www.anzdata.org.au/v1/index.html

\textsuperscript{2}http://archive.ics.uci.edu/ml/datasets/

\textsuperscript{3}http://www.cc.gatech.edu/gvu/user_surveys/
November 16, 1997. We extract a subset of the dataset for experiment; the extracted dataset has 10,181 entries (rows) and 9 attributes (columns).

A matrix of the various types of case studies associated with each dataset appears in Table 3.1.

<table>
<thead>
<tr>
<th>Data</th>
<th>Chap.</th>
<th>Ontology</th>
<th>Unexpectedness</th>
<th>Explanation</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANZDATA</td>
<td>4,7</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>Evaluation</td>
</tr>
<tr>
<td>DCT</td>
<td>7</td>
<td></td>
<td>√</td>
<td></td>
<td>Evaluation</td>
</tr>
<tr>
<td>KSL</td>
<td>5</td>
<td></td>
<td></td>
<td>√</td>
<td>Comparison, Demonstration</td>
</tr>
<tr>
<td>Census Income</td>
<td>6</td>
<td></td>
<td></td>
<td>√</td>
<td>Comparison, Demonstration</td>
</tr>
<tr>
<td>WWW User Survey</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td>Comparison, Demonstration</td>
</tr>
</tbody>
</table>

Table 3.1: Reference table for datasets and their locations of case studies.

### 3.1.1 ANZDATA dataset

For the past 31 years, the Australia and New Zealand governments have funded a registry that records the incidence, prevalence and outcome of dialysis and transplant treatment for patients with end stage renal failure: the ANZDATA registry (Australia and New Zealand Dialysis and Transplant Registry). The ANZDATA registry contains a wealth of information on kidney disease treatment and progression in Australia and New Zealand renal disease patients. Interested readers may find how ANZDATA has been used by others (McDonald et al., 2002; Lim et al., 2005; Haysom et al., 2009).

The dataset, a snapshot from the database, records some 96 attributes on each patient, including demographic, clinical and physiological data, on a yearly basis; 19,219 patients are represented for a total of 217,803 records. The 96 variables can be categorized into 7 types: record tags, clinical attributes, clinical findings, procedure, disease/syndrome, behavior, or health care activity (as listed in Tables 3.2 and 3.3). In the data, the median age at commencement
of dialysis and transplant treatment was 58.12 years, and the age range 0 to 97 years; 57% patients were male and the other 43% were female. 23.7% of patients had diabetic nephropathy attributed as their cause of end stage renal failure, 31.2% had glomerulonephritis, 11.5% renovascular disease, and 6.5% polycystic kidney disease. Additionally, as each patient is likely to have multiple entries, further information is implicitly encoded in the progression of time variant attributes. The dataset also contains a large percentage of missing values. Among all the 217,803 records of 19,219 patient, 44% entries are either missing or ‘not available’ and 13.6% entries are time varying.

The attributes in the dataset are either categorical or numerical, and some of the categorical attributes encode rich terminologies, and potentially knowledge, in their values: for example, the allowable values of the TreatmentModality attribute are: [1=Haemodialysis; 2=Peritoneal dialysis; 3=Transplant(Graft); 4=Return of native function; 5=Loss to follow up]. A fragment of the values in ANZDATA is shown in Table 3.4.

3.1.2 DCT dataset

The decimal comparison test (DCT) data, collected during 1995 to 1997, consists of test results from 3531 Australian students in grades 5 to 10, recording students’ answers to 30 decimal comparison items (see Section 7.2 for some background of the test). The data contains 7 item types where an item type is a set of comparison items that should be answered identically, either all correct or incorrect, by any given student who is consistently applying their own way of thinking about decimals. The item types and items (questions) are listed in Table 3.5. Each item type is designed to probe a particular decimal misconception; for instance the Type 1 items are designed for students who think the shorter after the decimal point is bigger. The DCT dataset has been analyzed in earlier work (Steinle and Stacey, 2003, 2004) and further used for intelligent tutoring systems (Nicholson et al., 2001; Boneh et al., 2004)

About a dozen mutually exclusive misconceptions have been identified from these data by domain experts. Table 3.6 shows the set of rules that the domain experts originally used to classify students based on their response to 6 types of DCT test items: High = High number correct (e.g. 4 or 5 out of 5), Low = Low number correct (e.g. 0 or 1 out of 5), with ‘.’ indicating that any performance level is observable for that item type by that class of students
### 3.1. MATERIALS

Table 3.2: Variables of ANZDATA (Part I).

<table>
<thead>
<tr>
<th>Class</th>
<th>Attribute(s)</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record Tag</td>
<td>roberts.id</td>
<td>Individual identification number.</td>
</tr>
<tr>
<td></td>
<td>merge</td>
<td>Results of latest merge.</td>
</tr>
<tr>
<td></td>
<td>sequence</td>
<td>Sequence of entries.</td>
</tr>
<tr>
<td></td>
<td>surveydate</td>
<td>Date of entry of information.</td>
</tr>
<tr>
<td></td>
<td>total</td>
<td>Total number of entries.</td>
</tr>
<tr>
<td></td>
<td>agedeath,ageabsdeath</td>
<td>Age at death (2 variables).</td>
</tr>
<tr>
<td></td>
<td>agequart</td>
<td>Quartiles of age for patients.</td>
</tr>
<tr>
<td></td>
<td>agestart</td>
<td>Age at first dialysis.</td>
</tr>
<tr>
<td></td>
<td>bmib</td>
<td>Body mass index.</td>
</tr>
<tr>
<td></td>
<td>dryweight,weightb</td>
<td>Weight in kg. (2 variables).</td>
</tr>
<tr>
<td></td>
<td>esrdstart</td>
<td>Date of commencement of renal replacement therapy.</td>
</tr>
<tr>
<td></td>
<td>height</td>
<td>Height at entry in meters.</td>
</tr>
<tr>
<td></td>
<td>race,raceb,raceothr</td>
<td>Race origin (3 variables).</td>
</tr>
<tr>
<td></td>
<td>referral,referralb</td>
<td>Late referral (2 variables)</td>
</tr>
<tr>
<td></td>
<td>sex</td>
<td>Gender.</td>
</tr>
<tr>
<td></td>
<td>alldeath</td>
<td>All-cause mortality.</td>
</tr>
<tr>
<td></td>
<td>cancdiag,cancdiagb</td>
<td>Cancer ever diagnosed (2 variables).</td>
</tr>
<tr>
<td></td>
<td>carddeath</td>
<td>Cardiac death.</td>
</tr>
<tr>
<td></td>
<td>causdeth,caudethb,dethothr</td>
<td>Cause of death (3 variables).</td>
</tr>
<tr>
<td></td>
<td>circdeath</td>
<td>Circulatory death.</td>
</tr>
<tr>
<td></td>
<td>cvdeath</td>
<td>Cardiovascular death.</td>
</tr>
<tr>
<td></td>
<td>disease,diseaseb,disothr</td>
<td>Cause of ESRD (3 variables).</td>
</tr>
<tr>
<td></td>
<td>graftsus,graftsusb</td>
<td>Graft function at time of death (2 variables).</td>
</tr>
<tr>
<td></td>
<td>failcard</td>
<td>Death from cardiac causes.</td>
</tr>
<tr>
<td></td>
<td>failcirc</td>
<td>Death from circulatory causes.</td>
</tr>
<tr>
<td></td>
<td>failcvd</td>
<td>Death from CVD causes.</td>
</tr>
<tr>
<td></td>
<td>failhf</td>
<td>Death from heart failure.</td>
</tr>
<tr>
<td></td>
<td>failmi</td>
<td>Death from myocardial infarction.</td>
</tr>
<tr>
<td></td>
<td>ureat</td>
<td>Urea reduction ratio at time of survey.</td>
</tr>
<tr>
<td></td>
<td>datbirth</td>
<td>Date of birth.</td>
</tr>
<tr>
<td></td>
<td>dod</td>
<td>Date of death</td>
</tr>
</tbody>
</table>

Of the 3531 tests (students) in total, exactly 1200 students were correct on
Table 3.3: Variables of ANZDATA (Part II).

<table>
<thead>
<tr>
<th>Class</th>
<th>Attribute(s)</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Procedure</td>
<td>biopsy,biopsyb</td>
<td>Renal biopsy (2 variables).</td>
</tr>
<tr>
<td></td>
<td>firstaccess,firstaccessb</td>
<td>Dialysis access used at first haemodialysis (2 variables).</td>
</tr>
<tr>
<td></td>
<td>flowrate,flowratet</td>
<td>Dialysis pump speed in ml/min at time of survey.</td>
</tr>
<tr>
<td></td>
<td>frequencyt</td>
<td>Dialysis sessions per week.</td>
</tr>
<tr>
<td></td>
<td>hourspwt,hourst</td>
<td>Dialysis hours per week at time of survey.</td>
</tr>
<tr>
<td></td>
<td>ktvhdt</td>
<td>Dialysis Kt/V at time of survey.</td>
</tr>
<tr>
<td></td>
<td>ktvpdt</td>
<td>Peritoneal dialysis Kt/V at time of survey.</td>
</tr>
<tr>
<td></td>
<td>present,presentt</td>
<td>Current dialysis access.</td>
</tr>
<tr>
<td></td>
<td>pett</td>
<td>PET test at time of survey.</td>
</tr>
<tr>
<td>Disease or Syndrome</td>
<td>cadb,cadsb,coronary</td>
<td>Coronary artery disease (3 variables).</td>
</tr>
<tr>
<td></td>
<td>cvd,cvdb,cvdsb</td>
<td>Cerebrovascular disease (3 variables).</td>
</tr>
<tr>
<td></td>
<td>diabetes,diabetesb</td>
<td>Diabetes (2 variables including type I and II).</td>
</tr>
<tr>
<td></td>
<td>diabetes1</td>
<td>Type I diabetes.</td>
</tr>
<tr>
<td></td>
<td>diabetes2</td>
<td>Type II diabetes.</td>
</tr>
<tr>
<td></td>
<td>diabetesi</td>
<td>Diabetes, insulin requiring</td>
</tr>
<tr>
<td></td>
<td>ht,hypertension</td>
<td>Hypertension requiring treatment (2 variables).</td>
</tr>
<tr>
<td></td>
<td>lung,lungb,lungsb</td>
<td>Chronic lung disease (3 variables).</td>
</tr>
<tr>
<td></td>
<td>pvd,pvdb,pvdsb</td>
<td>Peripheral vascular disease (3 variables).</td>
</tr>
<tr>
<td></td>
<td>vascdisb,vascdisb</td>
<td>Any vascular disease (2 variables).</td>
</tr>
<tr>
<td>Behavior</td>
<td>cig,ciga,cigc</td>
<td>Smoking (3 variables).</td>
</tr>
<tr>
<td>Activity</td>
<td>modality</td>
<td>Treatment modality.</td>
</tr>
<tr>
<td></td>
<td>likelihood</td>
<td>Likelihood of transplant.</td>
</tr>
<tr>
<td></td>
<td>txwlcat,waiting</td>
<td>Transplant waiting list status (2 variables).</td>
</tr>
</tbody>
</table>

every item (FineCode=ATE). In other words, the 2331 remaining students had at least one error. In total, the DCT dataset has 40 attributes - namely Year (grade of a student), FineCode, CoarseCode, Type1 to TypeS (7 attributes), Q1 to Q30 (30 attributes).
3.1. MATERIALS

<table>
<thead>
<tr>
<th>Item Type</th>
<th>Items (Questions)</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type 1 (Core)</td>
<td>Q6/Q7/Q8/Q9/Q10</td>
<td>4.8/4.63; 0.75/0.8</td>
</tr>
<tr>
<td>Type 2 (Core)</td>
<td>Q16/Q17/Q18/Q19/Q20</td>
<td>5.62/5.736; 0.426/0.3</td>
</tr>
<tr>
<td>Type 3 (Non-Core)</td>
<td>Q12/Q13/Q14/Q15</td>
<td>4.08/4.7; 3.72/3.073</td>
</tr>
<tr>
<td>Type 4 (Non-Core)</td>
<td>Q21/Q22/Q23/Q24</td>
<td>4.4502/4.45; 17.353/17.35</td>
</tr>
<tr>
<td>Type 5 (Non-Core)</td>
<td>Q3/Q4/Q5</td>
<td>0.3/0.4; 1.85/1.84</td>
</tr>
<tr>
<td>Type 6 (Non-Core)</td>
<td>Q26/Q27/Q28</td>
<td>0.35/0.42; 2.186/2.954</td>
</tr>
<tr>
<td>Type S (Supplement)</td>
<td>Q1/Q2/Q11/Q25/Q29/Q30</td>
<td>0.86/1.3; 0.006/0.53</td>
</tr>
</tbody>
</table>

3.1.3 KSL dataset

The dataset of Danish 70-year elderly has 1083 samples(rows), and each sample has 9 attributes(columns). The meaning of its attributes are listed in Table 3.8. The KSL dataset is distributed with the DEAL Bayesian network package (Boettcher and Dethlefsen, 2003). Interested readers may find how KSL dataset has been used by others (Jaroszewicz and Simovici, 2004; Mascherini and Stefanini, 2005; Malhas and Aghbari, 2009).
Table 3.6: Response patterns, classified in attributes CoarseCode and FineCode, expected from students with different misconceptions.

<table>
<thead>
<tr>
<th>Coarse Class</th>
<th>Fine Class</th>
<th>Item type (with sample item)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>ATE</td>
<td>1  2  3  4  5  6</td>
</tr>
<tr>
<td></td>
<td>AMO</td>
<td>4.8 5.76 4.7 0.45 0.4 0.42</td>
</tr>
<tr>
<td></td>
<td>AU</td>
<td>4.63 5.62 4.08 0.45 0.3 0.35</td>
</tr>
<tr>
<td>L</td>
<td>LWH</td>
<td>High High High Low High</td>
</tr>
<tr>
<td></td>
<td>LZE</td>
<td>High High High Low High</td>
</tr>
<tr>
<td></td>
<td>LRV</td>
<td>High High High Low High</td>
</tr>
<tr>
<td></td>
<td>LU</td>
<td>High High High Low High</td>
</tr>
<tr>
<td>S</td>
<td>SDF</td>
<td>High Low High Low Low</td>
</tr>
<tr>
<td></td>
<td>SRN</td>
<td>High Low High Low Low</td>
</tr>
<tr>
<td></td>
<td>SU</td>
<td>High Low High Low Low</td>
</tr>
<tr>
<td>U</td>
<td>MIS</td>
<td>Low Low Low Low Low</td>
</tr>
<tr>
<td></td>
<td>UN</td>
<td>.  .  .  .  .  .</td>
</tr>
</tbody>
</table>

Table 3.7: Scores of the categorisation on item types.

<table>
<thead>
<tr>
<th>Item Type</th>
<th>Items (Questions)</th>
<th>Scores of</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>High</td>
</tr>
<tr>
<td>Type 1</td>
<td>Q6/Q7/Q8/Q9/Q10</td>
<td>4, 5</td>
</tr>
<tr>
<td>Type 2</td>
<td>Q16/Q17/Q18/Q19/Q20</td>
<td>4, 5</td>
</tr>
<tr>
<td>Type 3</td>
<td>Q12/Q13/Q14/Q15</td>
<td>3, 4</td>
</tr>
<tr>
<td>Type 4</td>
<td>Q21/Q22/Q23/Q24</td>
<td>3, 4</td>
</tr>
<tr>
<td>Type 5</td>
<td>Q3/Q4/Q5</td>
<td>2, 3</td>
</tr>
<tr>
<td>Type 6</td>
<td>Q26/Q27/Q28</td>
<td>2, 3</td>
</tr>
</tbody>
</table>

Table 3.8: Attributes of the KSL dataset.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Meaning</th>
<th>Attribute Type (Value)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FEV</td>
<td>Forced ejection volume</td>
<td>Continuous (38 ~ 378)</td>
</tr>
<tr>
<td>Kol</td>
<td>Cholesterol</td>
<td>Continuous (288 ~ 1344)</td>
</tr>
<tr>
<td>BMI</td>
<td>Logarithm of Body Mass Index</td>
<td>Continuous (15.8 ~ 46.9)</td>
</tr>
<tr>
<td>Hyp</td>
<td>Hypertension</td>
<td>Nominal (no,yes)</td>
</tr>
<tr>
<td>Smok</td>
<td>Smoking</td>
<td>Nominal (no,yes)</td>
</tr>
<tr>
<td>Alc</td>
<td>Alcohol consumption</td>
<td>Nominal (seldom,frequently)</td>
</tr>
<tr>
<td>Work</td>
<td>Working</td>
<td>Nominal (yes,no)</td>
</tr>
<tr>
<td>Sex</td>
<td>Gender</td>
<td>Nominal (male,female)</td>
</tr>
<tr>
<td>Year</td>
<td>Survey year</td>
<td>Nominal (1967,1984)</td>
</tr>
</tbody>
</table>
3.2. METHODS

3.1.4 Census Income dataset

The Census Income (or called Adult) dataset was extracted from the 1994 US Census database as an exemplar dataset for developing predictive models of income based on census data; the dataset is downloadable from the UCI Machine Learning Repository\(^4\). The dataset has 14 variables - either categorical or numerical; they are age, work class, education, education number, marital status, occupation, relationship, race, gender, capital gain, capital loss, working hours per week, native country and income; the data has 32,562 instances and their values are listed in Table 3.9. Interested readers may find how the dataset has been used by others (Frank et al., 2002; Bay, 2001; Oza and Russell, 2001).

3.1.5 WWW User Survey dataset

This dataset is derived from the Graphic, Visualization, and Usability Center’s (GVU) 8th WWW User Survey\(^5\) which is available online for analysis. The survey was run from October 10, 1997 through November 16, 1997. Over 10,000 web users participated in the survey; questions were asked on the following areas: general demographics, technology demographics, data privacy, web and internet usage, internet shopping, information gathering and purchasing, opinions on internet commerce, and etc. The original dataset has 10,181 instances and 71 variables. The dataset used in this research consists of a subset of 10 of the 71 variables related to users’ willingness to pay for content on the web; the selected variables are listed in Table 3.10

3.2 Methods

In the thesis, three aspects of KDD are researched: ontology-driven DM, unexpectedness, and explanation generation. We have access to one clinical nephrologist in mining the ANZDATA dataset, and access to two educational experts in mining the DCT dataset; the case studies of the two datasets are for

\(^4\)http://archive.ics.uci.edu/ml/datasets/Adult

\(^5\)http://www.cc.gatech.edu/gvu/user_surveys/survey-1997-10/
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>17 ∼ 90</td>
</tr>
<tr>
<td>Education</td>
<td>Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.</td>
</tr>
<tr>
<td>Education number</td>
<td>1 ∼ 16</td>
</tr>
<tr>
<td>Relationship</td>
<td>Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.</td>
</tr>
<tr>
<td>Race</td>
<td>White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.</td>
</tr>
<tr>
<td>Gender(Sex)</td>
<td>Female, Male.</td>
</tr>
<tr>
<td>Capital gain</td>
<td>0 ∼ 99999</td>
</tr>
<tr>
<td>Capital loss</td>
<td>0 ∼ 4356</td>
</tr>
<tr>
<td>Working hours per week</td>
<td>1 ∼ 99</td>
</tr>
<tr>
<td>Native country</td>
<td>United States, Cambodia, England, Puerto Rico, Canada, Germany, Outlying US(Guam USVI etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El Salvador, Trinidad and Tobago, Peru, Hong, Holand Netherlands.</td>
</tr>
<tr>
<td>Income</td>
<td>50K, ≤ 50K.</td>
</tr>
</tbody>
</table>

evaluation purpose; together, we use the five datasets to demonstrate, compare or evaluate the proposed methods:

**Demonstrative Case Studies.** In terms of unexpectedness, we apply the KSL and Census Income datasets to test the behavior of the proposed design (Subsections 5.4.1 and 6.5.2). In terms of explanation generation, we apply the WWW User Survey and Census Income datasets to demonstrate the benefit of the presented method (Subsections 6.5.1 and 6.5.2). The case study of Census Income (Subsection 6.5.2) unifies the
Table 3.10: The selected questions/variables in the survey of Internet use.

<table>
<thead>
<tr>
<th>Question (attribute)</th>
<th>Examples of choice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual Job</td>
<td>Accountant; Administrator/Secretary; Advertising Professional; ...</td>
</tr>
<tr>
<td>Education Attainment</td>
<td>Grammar School; High School; College Graduate; ...</td>
</tr>
<tr>
<td>Gender</td>
<td>Female; Male</td>
</tr>
<tr>
<td>Household Income</td>
<td>Under $10,000; $10,000-$19,999; Over $100,000; ...</td>
</tr>
<tr>
<td>Major Occupation</td>
<td>Computer Related; Management; Professional; Educator and/or Student; Other</td>
</tr>
<tr>
<td>Marital Status</td>
<td>Divorced; Married; Single; ...</td>
</tr>
<tr>
<td>Web Ordering (ever ordered via web)</td>
<td>Yes; No; Don’t know</td>
</tr>
<tr>
<td>Web Page Creation</td>
<td>Yes; No; Don’t know</td>
</tr>
<tr>
<td>Willingness to Pay Fees</td>
<td>Cost too high; The information is of poor quality; Lack of easy payment mechanisms; ...</td>
</tr>
<tr>
<td>Years on Internet</td>
<td>Less than 6 months; 6 to 12 months; 1 to 3 years; ...</td>
</tr>
</tbody>
</table>

unexpectedness and explanation generation together to demonstrate how ‘explanations’ could be applied to unexpectedness.

Comparative Case Studies. We also make comparison of the demonstrative case studies to previous research. In Subsections 5.4.1 and 6.5.2 of the KSL and Census Income datasets, our method of unexpectedness is also compared with the method of Jaroszewicz and Simovici (2004). In terms of explanations, a comparison with the approach of Yao et al. (2003) is also made over the WWW User Survey dataset (Subsection 6.5.1).

Evaluative Case Studies. We apply the ANZDATA to the evaluation of the ontology-driven data mining. In terms of unexpectedness and explanation generation, the ANZDATA and DCT datasets are tested with the collaboration of domain experts.

3.3 Chapter Summary

There are five datasets used in the experiments described in this thesis: ANZDATA, DCT, KSL, Census Income, and WWW User Survey. In evaluative case studies, we worked with three domain experts who play the role of KDD-users:
one medical expert for ANZDATA and two educational experts for DCT. The summary of how the datasets are used in the thesis is listed in Table 3.1.
Chapter 4

Ontology-Driven Data Mining

In preliminary experiments on deriving knowledge from the ANZDATA dataset, we applied naïve association rules, decision trees, and emerging pattern mining techniques (Ramamohanarao and Bailey, 2003) to a subset of the data (in which a targeted subset of attributes were selected for a reduced set of records); the preliminary experiment is included in Appendix G. Unsurprisingly, the first problem we encountered is that an overwhelmingly large set of rules is generated, and furthermore many of the rules are either already known, or seem reasonable (that is, simple to explain in the context of the user’s knowledge) to a domain expert (in this case, a senior nephrologist). In other words, uninformed data mining was generally unable to provide interesting or surprising patterns to the expert, perhaps because these patterns were obscured by the sheer number of rules.

Further, we noticed a category of problem which we will call the ‘meaningless’ problem (this term was originally coined by the user). The ‘meaningless’ problem is caused by either trivial, uninformative, or nonsensical rules; note that the ‘meaningless’ impression is purely subjective to the user only, even though the rules may still be valuable from another point of view. Examples of some meaningless rules are:

**Trivial rules.** DieOfCancer $\rightarrow$ ¬DieOfCardiacDisease is an example of trivial rules as we know the causes of death are mutually exclusive.

**Uninformative rules.** Weight$=53 \sim 101kg \rightarrow$ DieOfCardiacDisease would be uninformative because the expert has to reason weight together with age, gender, and height. This problem frequently appeared in our case studies (reported in Chapter 7) and is further analysed in Subsection 7.3.4.
**Nonsensical rules.** NoCancer, Deceased → TotalRecords>5 would be regarded as nonsensical because a patient’s total number of records is essentially an artefact of the data collection system, not meaningful to the domain expert/user (although there may be some relation between the age/length of treatment and number of records, the expert considers the specific rule to be nonsensical).

In addition to the three types of meaningless rules, there is another type of rules which we called it as ‘pointless’ rules; for example Diabetes=Type1 → Type1Diabetes=Yes is a pointless rule caused by duplicate variables. Because avoiding/removing duplicate variables is the basic requirement in data selection/preprocessing, this problem has not bothered us in the case studies. We would assume that the data is well preprocessed so that no ‘pointless’ rules would appear; thus the ‘pointless’ problem will not be discussed further.

If a rule is meaningless (to the user), we might get some feedback like this: ‘Well, I am not sure whether the rule is surprising or not because it does not make sense to me.’ As a consequence, meaningfulness (which is subjective and hard to define), should be one prerequisite before identification of surprising rules. Therefore, in attempting to tackle this problem, the first part of the chapter discusses an investigation of utilising a domain ontology to drive the mining of association rules in an effort to eliminate meaningless results.

The second part of this chapter attempts to study how to exploit domain ontologies for surprisingness. In search of an efficient way to acquire a user’s knowledge, an alternative to requiring a domain expert to populate a KB might be to use an established KB as the source of DK, and therefore avoid the knowledge acquisition bottleneck. A domain ontology might be considered such a KB; the premise of this idea relies on the assumption that the expert’s knowledge is consistent with the ontology. For example, the Unified Medical Language System (UMLS) is a medical ontology which is maintained by the National Library of Medicine in the U.S. (Lindberg, 1990); we assume that a medical expert’s knowledge is consistent with the UMLS because it is formally collected by the authority. In other words, the premise implies that if an ontology contains sufficient information and it is identical with a user’s knowledge, it can be used as the KB.

Based on the idea of reducing or eliminating the knowledge acquisition bottleneck, we investigate the possibility of incorporating a domain ontology into data mining for discovering surprising patterns. There are two questions
in this investigation: (1) How can an ontology be utilised for the mining of surprising patterns? (2) What kind of information should be included in the ontology?

The chapter proceeds as follows: Section 4.1 formalises a design for addressing the issue of ‘meaningfulness’ of rules. Section 4.1.2 reports a complete study of applying the first design to the ANZDATA dataset. Section 4.2 proposes a hypothetical design of integrating an ‘ontology knowledgebase’ to mine unexpected patterns; the second design responses the question of essential information of ontology knowledgebases.

4.1 Design I - Ontology for Semantic-Rich Association Rules

In addressing the meaningfulness problem, we follow the idea of (Cespivova et al., 2004) that an ontology can be used as an ‘ontological bias’ to provide semantic structures for association rules. In the design, we formalise this idea with some modification. There are two common components of ontologies needed in this design - namely individuals and classes. An individual is an instance or an object; for example, Diabetes is an individual in a medical ontology. A class is a collection of concepts (individuals); for instance, Disease is a class of \{Diabetes, Cancer, Tuberculosis\}.

Generally, the idea is that the attributes of a dataset can be aligned with the individuals in the ontology. Then we can further categorise the attributes into corresponding classes. For example, suppose Renal Biopsy is an attribute in a dataset; it is aligned with Kidney Biopsy in the UMLS; the corresponding class is Diagnostic Procedure. Figure 4.1 shows the idea of mapping attributes to ontology individuals and corresponding classes.

On top of the ontological mapping discussed above, semantic structures can be defined over the classes of an ontology in the format of association. For example, the semantic structure Behavior $\rightarrow$ Disease or Syndrome defines a template for association rules which contains attributes in class Behavior on
CHAPTER 4. ONTOLOGY-DRIVEN DATA MINING

Figure 4.1: Diagram of the mapping from attributes to classes of an ontology.

LHS\(^1\) and Disease or Syndrome on RHS\(^2\). The semantic structures provide a means for a user to define meaningful templates for association rules.

**Definition 1** (Semantic structures). A semantic structure has the format \(X \rightarrow Y\), where \(X\) and \(Y\) are the classes of an ontology.

4.1.1 Overall Procedure

In the procedure, an ontology is used to categorise attributes into classes; then the semantic structures are defined over the classes; finally, association rules are mined under the guidance of the semantic structures.

1. **Attribute Alignment.** Align each attribute with its corresponding individual in an ontology. This is done manually, but potentially may be addressed by natural language processing (NLP).

2. **Individual Categorisation.** Map each individual into its related class. This can be done by directly querying the ontology.

3. **Semantic Structure.** Assign some preferred semantic structures over the classes which are obtained from the previous step. A straightforward implementation of this step would be to use input from the user; again, approaches from NLP may also be possible.

4. **Association Mining.** Finally, mine association rules under the constraint of the semantic structures.

\(^1\)Left hand side (of a rule).

\(^2\)Right hand side (of a rule).
4.1.2 Case Study: Nephrology

In this case study, we take a well known ontology and mine and rank association rules in order to derive knowledge from the ANZDATA dataset; this knowledge is then evaluated against common knowledge in kidney disease. The medical expert provided two commonly agreed associations in chronic kidney disease and asked whether the associations could be re-discovered by data mining; the two associations are:

- Diabetes (type1,type2) $\rightarrow$ CardiovascularDeath
- Smoking $\rightarrow$ CardiovascularDisease $\rightarrow$ CardiovascularDeath

The Medical Ontology

The ontology we use in this case study is the Unified Medical Language System\(^3\) (UMLS), a medical ontology developed by the National Library of Medicine of U.S. The UMLS is a compendium of many controlled vocabularies in the biomedical science. It provides a mapping structure among these vocabularies and thus allows one to translate among the various terminology systems; it may also be viewed as a comprehensive thesaurus and ontology of biomedical concepts. UMLS consists of the following components:

**Metathesaurus.** The core database of the UMLS, a collection of concepts and terms from the various controlled vocabularies, and their relationships.

**Semantic Network.** A set of categories and relationships that are being used to classify and relate the entries in the Metathesaurus. Figure 4.2 shows and example of the Semantic Network in UMLS.

**SPECIALIST Lexicon.** A database of lexicographic information for use in natural language processing.

\(^3\)\url{http://www.nlm.nih.gov/research/umls/}
Attribute Alignment and Individual Categorization

There are 96 attributes of which 11 attributes have no definitions. The remaining 85 defined attributes are mapped to their corresponding classes via the help of UMLS (Tables 3.2 and 3.3). Six of the seven classes are defined in the UMLS. They are ‘Clinical Attribute,’ ‘Finding,’ ‘Procedure,’ ‘Disease or Syndrome,’ ‘Behavior,’ and ‘Health Care Activity.’ The ‘Record Tag’ group is not recorded in UMLS but essential for data preprocessing and transforming. The semantic type ‘Finding’ is defined as that which is discovered by direct observation or measurement of an organism attribute or condition, including the clinical history of the patient.

Semantic Structures

Based on the classes listed in Tables 3.2 and 3.3, we define 7 semantic structures for association rules:

- Disease or Syndrome $\rightarrow$ Finding(Death)
- Disease or Syndrome $\rightarrow$ Disease or Syndrome
- Clinical Attribute $\rightarrow$ Disease or Syndrome
- Finding(except Death) $\rightarrow$ Disease or Syndrome
- Procedure $\rightarrow$ Disease or Syndrome
- Behavior $\rightarrow$ Disease or Syndrome
4.1. DESIGN I - ONTOLOGY FOR SEMANTIC-RICH ASSOCIATION RULES

- Health Care Activity $\rightarrow$ Disease or Syndrome

Cascaded Association Rules

The experiment devises an approach to cascade association rules together because one of the target association of evaluation has three layers. For example, if there are two rules: Obese $\rightarrow$ Diabetes and Diabetes $\rightarrow$ CardiovascularDeath, it would be concise to represent Obese $\rightarrow$ Diabetes $\rightarrow$ CardiovascularDeath.

Instead of devising a new algorithm for cascading mined association rules, we take a simple approach in the consideration of performance. If we select a particular attribute as the RHS attribute, we can tailor the search of association rules focusing on the attribute. For example, if attribute B is selected, then the search space is reduced to find every attribute A so that A $\rightarrow$ B is valid; the search of A costs $O(n)$ where n is the number of attributes. Combined with the idea of semantic structures, the search space could be greatly reduced.

In the formation of a reasoning chain, the next step is to repeat the search of association rules that finds every attribute X satisfying X $\rightarrow$ A, where A is discussed in the previous example; therefore a cascaded association rule is found to be X $\rightarrow$ A $\rightarrow$ B. Notably, this approach is suitable for one-attribute-rules only.

Interestingness Measure

In order to rank rules more closely aligned with the user’s interest, we formulate a new interestingness measure which is inspired by (Rauch, 2002). Ohsaki et al. (2007) have evaluated the usefulness of 40 objective measures for medical KDD, concluding that the performance of interestingness measures is influenced by the certainty of a hypothesis made by medical experts. In the experiment, according to the expert’s preference, the proposed measure is based on the feature of risk-factor metric in medicine.

The association rule (AR) is here understood as an expression A $\rightarrow$ S where A and S are single or multiple attribute-value pairs; For example, A=(Hypertension=Yes, Diabetes=Yes). The rule A $\rightarrow$ S can be interpreted as ‘The occurrence of A is highly associated with the occurrence of S.’

In defining the new measurement, a four-fold table is introduced in Table 4.1. Here a is the number of entries satisfying both A and S, b is the number
of entries satisfying \( A \) but not satisfying \( S \), \( c \) and \( d \) are defined in the similar way. The confidence of the rule \( A \rightarrow S \) is \( P(S|A) = \frac{a}{a+b} \), and the confidence of the rule \( \neg A \rightarrow S \) is \( P(S|\neg A) = \frac{c}{c+d} \). The new measurement, ‘Compared Confidence’, measures the proportion of the difference between \( A \rightarrow S \) and \( \neg A \rightarrow S \).

**Definition 2** (Compared Confidence). The compared confidence, denoted by \( \theta \), is defined by the following equation.

\[
\theta = \frac{P(S|A) - P(S|\neg A)}{P(S|\neg A)}.
\] (4.1)

If an association rule’s compared implication is above a threshold, it is denoted as \( A \rightarrow S \mid_{\theta} \).

The interpretation of \( A \rightarrow S \mid_{\theta} \) could be ‘the probability of \( P(S \mid A) \) is \((100 \times \theta)\% \) times higher than \( P(S \mid \neg A) \).’ The compared confidence is designed to highlight a rule’s confidence against its opposite so that the uniform distribution of \( S \) over \( A \) can be ruled out.

Table 4.1: Four-fold table for data analysis.

<table>
<thead>
<tr>
<th>4ft</th>
<th>( S )</th>
<th>( \neg S )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A )</td>
<td>( a )</td>
<td>( b )</td>
</tr>
<tr>
<td>( \neg A )</td>
<td>( c )</td>
<td>( d )</td>
</tr>
</tbody>
</table>

**Example**

Table 4.2 shows a pseudo data-based explanation of idea of \( \theta \). This data satisfies \( \text{support} = 45\% \) and \( \text{confidence} = 90\% \) for the rule \( A \rightarrow S \); Whilst this pseudo data is significant under other measurements, e.g. \( \text{confidence} = 90\% \), ‘\( \text{Founded Implication} \)’ = 0.9 and ‘\( \text{Double Founded Implication} \)’ \( \approx 0.47 \) in (Rauch, 2002), \( S \) is actually evenly distributed so that a user would not think the rule is significant. In this case, the proposed compared confidence \( \theta = 0 \) means that this data is not significant.

There are, of course, situations that lead to \( \theta < 0 \) if \( P(S|A) < P(S|\neg A) \). However, it would not be a problem if we focus on the rules with \( \text{confidence} > 50\% \).

Because ‘\( \text{risk factor} \)’ is widely used in the medical domain and is concise, the case study implementation focusses on finding association rules with only
4.1. DESIGN I - ONTOLOGY FOR SEMANTIC-RICH ASSOCIATION RULES

Table 4.2: Four-fold table of a pseudo data.

<table>
<thead>
<tr>
<th>[4ft]</th>
<th>S</th>
<th>[\neg S]</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>[\neg A]</td>
<td>9</td>
<td>1</td>
</tr>
</tbody>
</table>

one literal on the left hand so that the mined rules would appear more familiar to a domain expert. Although the compared confidence of association rules, \(\theta\), is different from the comparable ‘attributable risk’ measurement which is used in epidemiology, they share a similar statistical sense; ‘attributable risk’ is defined as \(P(S|A) - P(S|\neg A)\). The succeeding paragraph will use the term ‘risk factor’ to denote the LHS of an association rule.

**Result**

There are 4 risk factors found for cardiovascular death (Table 4.3). The interpretation for the first risk factor is, ‘For patients who are already deceased, those who have never been diagnosed with cancer are 76% times higher to die of cardiovascular disease than those who have ever diagnosed with cancer.’ Similarly, the risk factors coronary artery disease, diabetes and hypertension are 18%, 17%, 16% more prevalent for cardiovascular death.

The full association tree is shown in Figure 4.3. The leftmost part is the risk factors of ‘no cancer’, ‘coronary artery disease’, ‘diabetes’, and ‘history of hypertension’ respectively. From top to bottom, the risk factors are sorted in descending order according to \(\theta\). Notably, this result successfully identifies the two known associations in chronic kidney disease: Diabetes \(\rightarrow\) CardiovascularDeath; Smoking \(\rightarrow\) CardiovascularDisease \(\rightarrow\) CardiovascularDeath.

The other two risk factors, other than the ones of evaluation, was also explained by the expert. In terms of ‘no cancer’, a reasonable explanation is that when people have cancer, they usually die of cancer; otherwise, cardiovascular death is the major cause of death for kidney disease patients. The fourth factor, ‘history of hypertension’, is also a common risk factor of cardiovascular death, added by the expert.

For comparison, conventional association mining is also performed on the data. The result is shown in Table 4.4. The first risk factor, ‘Circulatory death’, is a synonym of cardiovascular death and the second risk factor, ‘Death
from heart failure', is co-existing with cardiovascular death. The obesity re-
related findings: ‘Base weight’, ‘Body mass index’, and ‘Weight at entry’ are 
not directly related to cardiovascular death. Actually obesity is a risk factor 
of some diseases such as diabetes (type 2).

Table 4.3: The result of risk factors of cardiovascular death with semantic 
structures.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
<th>$\theta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancer ever diagnosed</td>
<td>No</td>
<td>0.76</td>
</tr>
<tr>
<td>Coronary artery disease</td>
<td>Yes</td>
<td>0.18</td>
</tr>
<tr>
<td>Diabetes</td>
<td>Yes</td>
<td>0.17</td>
</tr>
<tr>
<td>Hypertension</td>
<td>History of hypertension</td>
<td>0.16</td>
</tr>
</tbody>
</table>

Table 4.4: The result of risk factors of cardiovascular death without semantic 
structures.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
<th>$\theta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circulatory death</td>
<td>Circulatory system death</td>
<td>23.01</td>
</tr>
<tr>
<td>Death from heart failure</td>
<td>Death due to HF</td>
<td>1.07</td>
</tr>
<tr>
<td>Cancer ever diagnosed</td>
<td>No</td>
<td>0.76</td>
</tr>
<tr>
<td>Dialysis access used at first</td>
<td>Non-tunnel CV Catheter</td>
<td>0.43</td>
</tr>
<tr>
<td>haemodialysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Graft functioning at time of death</td>
<td>Not with functioning graft</td>
<td>0.41</td>
</tr>
<tr>
<td>Base weight</td>
<td>149.9~198.2 (kg)</td>
<td>0.32</td>
</tr>
<tr>
<td>Dialysis pump speed</td>
<td>350.8~425.4 (ml/min)</td>
<td>0.29</td>
</tr>
<tr>
<td>Racial origin</td>
<td>Maori</td>
<td>0.28</td>
</tr>
<tr>
<td>Body mass index</td>
<td>51.5~63.9</td>
<td>0.28</td>
</tr>
<tr>
<td>Weight at entry</td>
<td>114.1~151.1 (kg)</td>
<td>0.26</td>
</tr>
<tr>
<td>Caused of ESRD</td>
<td>Diabetes</td>
<td>0.2</td>
</tr>
<tr>
<td>Likelihood of transplant</td>
<td>24.8~39.7</td>
<td>0.19</td>
</tr>
<tr>
<td>Coronary artery disease</td>
<td>Yes</td>
<td>0.18</td>
</tr>
<tr>
<td>Body mass index</td>
<td>39.1~51.5</td>
<td>0.17</td>
</tr>
<tr>
<td>Diabetes</td>
<td>Yes</td>
<td>0.17</td>
</tr>
<tr>
<td>Any vascular disease</td>
<td>Yes</td>
<td>0.16</td>
</tr>
<tr>
<td>Hypertension</td>
<td>History of hypertension</td>
<td>0.16</td>
</tr>
<tr>
<td>Racial origin</td>
<td>Aborigine or TSI</td>
<td>0.16</td>
</tr>
<tr>
<td>Dialysis hours per session</td>
<td>7.0~8.5 (hr)</td>
<td>0.16</td>
</tr>
<tr>
<td>Diabetes, Type 2</td>
<td>Yes</td>
<td>0.15</td>
</tr>
</tbody>
</table>
4.1. DESIGN I - ONTOLOGY FOR SEMANTIC-RICH ASSOCIATION RULES

Figure 4.3: Result of mining risk factors of cardiovascular death
Discussion

This case study demonstrates the value of the use of an ontology in guiding data mining via semantic structures. The result passes the evaluation criteria that it successfully re-discovered known associations in kidney disease.

A comprehensive analysis of the issues encountered in mining interesting association rules was done, and here it is listed in Table 4.5. Most of the problem descriptions are self-explanatory, but several deserve more detailed explanation. The third listed problem in the ‘Transforming’ step - ‘Terminologies within entries’ - occurs when the values of a variable are specific medical terminologies. For example, the variable ‘disease’ has entries: ‘diabetes, polycystic kidney disease, renovascular disease, reflux nephropathy’ and etc. Because the process of variable categorization is performed for variables, we need these kind of entries to be appended as new variables. Potentially, we suggest ontology may be applied to some of the issues in Table 4.5, such as solving duplicate meaning among variables, discriminating irrelevant variables in data preprocessing step, identifying terminologies within entries and identifying synonyms in the result interpretation step.

Table 4.5: Some of the problems encountered during mining ANZDATA.

<table>
<thead>
<tr>
<th>Step</th>
<th>Problem</th>
<th>Ad hoc solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preprocessing</td>
<td>The concept of a patient</td>
<td>Identification number</td>
</tr>
<tr>
<td></td>
<td>Multiple records of a patient</td>
<td>aggregate</td>
</tr>
<tr>
<td></td>
<td>Duplicate meaning among variables</td>
<td>remove; merge;</td>
</tr>
<tr>
<td></td>
<td>Irrelevant variables</td>
<td>remove</td>
</tr>
<tr>
<td>Transforming</td>
<td>Numerical data</td>
<td>Discretisation</td>
</tr>
<tr>
<td></td>
<td>Meaningful range of Discretisation</td>
<td>Domain knowledge</td>
</tr>
<tr>
<td></td>
<td>Terminologies within entries</td>
<td>Append as new variables</td>
</tr>
<tr>
<td></td>
<td>Time varying variables</td>
<td>Aggregate</td>
</tr>
<tr>
<td>Data Mining</td>
<td>Data format</td>
<td>Data Transforming</td>
</tr>
<tr>
<td></td>
<td>Choice of DM algorithm</td>
<td>DM expertise</td>
</tr>
<tr>
<td></td>
<td>Parameters tuning</td>
<td>DM expertise</td>
</tr>
<tr>
<td>Interpretation</td>
<td>Meaningless finding</td>
<td>Refine input data</td>
</tr>
<tr>
<td></td>
<td>Synonymous</td>
<td>Refine input data</td>
</tr>
</tbody>
</table>

In summary, this study demonstrates how data mining could be driven by a medical ontology. The use of a medical ontology in specifying the semantic
4.2 Design II - Ontology for Unexpected Rules

In the second design, we address the question of mining unexpected rules with help from a formal ontology. This is a hypothetical design because we cannot conduct actual experiments due to the insufficiency of the ontology we have at hand. However, the design still provides an answer to the second question of the investigation: what kind of information should be embedded in the ontology.

In the design, we consider the possibility of building up a Bayesian network from a domain ontology. One common component of ontologies is relation; it describes ways in which classes and individuals can be related to one another. Extending the basic scope of relations, it is also possible to define relations among individuals. For example, related-to, associated-with, causes are possible relations over individuals. Therefore, if there exists individual-level relations in an ontology, it is possible to automatically map the structure of the ontology to a DAG (directed acyclic graph) of a Bayesian network.

Having an ontology derived Bayesian network, we then can apply the design of Jaroszewicz and Simovici (2004) to mine unexpected patterns. This seemingly simple idea suggests a potential way to start mining unexpected patterns with very little input from a user, as long as there is a suitable domain ontology. The sketch of the design is presented below.

1. Ontology to Bayesian Network. Extract all connections of the individuals of an ontology based on pre-selected relations. Then assign the edges of a graph directly from the extracted connections. Finally, adjust the orientations of edges to convert the graph into a DAG; build up a Bayesian network from the acquired DAG.
2. **Parameters Learning.** Learn the parameters of the Bayesian network from the data.

3. **Association Rules.** Perform association rules mining.

4. **Unexpected Rules.** Rank mined association rules based on the unexpectedness which uses the Bayesian network as KB.

5. **Iteration.** Update the Bayesian network according to a user’s interpretation of unexpected rules. Then perform unexpected rules mining again.

The requirement for applying this design is the existence of individual-level relations in an ontology base. In step 1, the connections of individual-level relations are used for building up the structure of BNs.

Although this design seems practical, currently there seem to be few existing suitable domain ontologies. Even if there exists such an ontology, individual-level relations might be absent: for example, there are no individual-level relations captured in the UMLS medical ontology which is explored in previous case study. In the end, the proposed design is unlikely to fit to most data mining projects unless ontology knowledgebases are widely developed in many domains with essential information captured.

**4.3 Discussion and Conclusion**

The first design of semantic-rich association rules mining partially resolves the ‘meaningless’ problem, but there are still other causes for ‘meaningless’ which are beyond the capability of the design; we will discuss them in Chapter 8. The second design for unexpectedness hypothetically proposes that domain ontologies could be a source of the knowledgebase in mining unexpected patterns, and the essential information for this aim is individual-level relations.

The major problem of incorporating domain ontologies into KDD is its availability: there is no ready for use domain ontologies for most KDD projects. One may argue that we could customise the ontology knowledgebase for a KDD project, but we doubt the return on effort if the ontology would not be reused or its utility could be replaced by other approaches. Given these difficulties, the thesis will not present further research on using domain ontologies in the
mining of surprising patterns, but rather will focus on mining surprising patterns using user-derived knowledgebases. In next chapter, we will present a design of unexpectedness that acquires KB from the user directly.
Chapter 5

Unexpected Pattern Mining

Unexpected pattern mining is the computational approach for the aim of finding surprising patterns (as justified in Chapter 1 and Section 2.2.1). In this chapter we present a design of unexpectedness; the design aims to provide a convenient interface of knowledge acquisition, and provide scalability to large datasets. In terms of the choice of knowledge representation, it is based on our review of unexpectedness (Subsection 2.4).

5.1 Motivation

In the previous chapter, we proposed the use of a domain ontology for unexpected pattern mining. However, the design requires a pre-existing ontology with essential information that would not exist in many data mining projects. Therefore, we now investigate the common approach of related work; that is to acquire knowledge from user(s).

In addressing the research question, the investigation was carried out with an empirical, cyclical approach - meaning that we first formalise a prototype for unexpectedness with a design for knowledge acquisition, then test the prototype with the help from domain experts; then, with the experience gained, we further analyse the problems encountered and propose solutions to them.

5.2 Problem Statement

Due to the nature of unexpectedness and it’s relationship to user’s knowledge, an interactive design of knowledge discovery is commonly used. Therefore, a user is at the center of the problem of surprisingness. On the one hand,
we have data and data-driven DM operated according to the user’s interest, and rules can be mined appropriately. On the other hand, we have some representation (KB) of the users’ knowledge (DK), and some approach to rule ranking based on both the KB and the unexpectedness measure. That is a simple but all-encompassing outline of the problem.

**Definition 3** (Unexpectedness Problem). Given a set of association (or dependence) rules\(^1\) \((R)\), mined from a dataset \((D)\), the problem of unexpectedness is to highlight a set of unexpected rules \((R_U)\) by an unexpectedness measure/function \((U)\):

\[
    r \in R_U \quad if \quad U(KB, r) > \epsilon,
\]

where \(KB\) is the knowledgebase acquired from the user, and \(\epsilon\) is the cutoff threshold for unexpectedness.

### 5.3 Proposed Design

In Chapter 2, we identified that probabilistic approaches have an advantage over other knowledge representations (e.g. rule based representations), in that they exhibit transitivity (transitive closure). Bayesian networks, one of the probabilistic representation, have been proposed as a suitable platform of knowledge representation for unexpectedness (Jaroszewicz and Simovici, 2004), and the design proposed here for mining surprising patterns focusses on devising algorithms for both unexpected association rules and unexpected dependence rules. The design includes several novel elements which have not been addressed before in unexpectedness; they are the independence test for rules, new design for knowledge acquisition, and scalability.

#### 5.3.1 Algorithms for Unexpectedness

We define a measure of unexpectedness in the context of conditional probability; it is different from the measure defined in (Jaroszewicz and Simovici, 2004). The reason why we use conditional probability (confidence) rather than

---

\(^1\)Introduced in Appendix B.
joint probability (support), is that conditional probability is a common concept in medicine (e.g. risk factors). For an association rule $A \rightarrow B$, we can infer the conditional probability from knowing the itemset $A$ to $P(B|A)$ by utilising a Bayesian network in a KB. The inferred conditional probability, $\hat{P}(B|A)$, serves as the computational expectation. Unexpectedness is defined on the difference between the confidence of a rule and its inferred conditional probability $\hat{P}(B|A)$.

**Definition 4** (Unexpectedness for Association Rules). Given a Bayesian network $BN$ and data $D$, the **unexpectedness** of an association rule $r$, $r = (A \rightarrow B)$, is defined as:

$$U(r) = \left| P_D(B|A) - \hat{P}_{BN}(B|A) \right|,$$

where $P_D$ is the relative frequency in data $D$, and $\hat{P}_{BN}$ is the conditional probability inferred from $BN$.

The algorithm of mining unexpected association rules is shown in Algorithm 1. The parameters of the local BN are estimated by the $\text{LearnParameters}(\cdot, \cdot)$ function.

**Input**: $D$: data, $BN_{local}$: local Bayesian network, $AR$: association rules, $\epsilon_U$: threshold for unexpectedness  

**Result**: $RU$: Unexpected rules  

$RU \leftarrow \emptyset$;  

forall $r \in AR$ do  

$BN \leftarrow BN_{local}(r)$ (by Algorithm 4);  

$\text{LearnParameters}(BN, D)$;  

$U(r) = \left| P_D(B|A) - \hat{P}_{BN}(B|A) \right|$;  

if $U(r) > \epsilon_U$ then  

$\text{Append } r \text{ to } RU$;  

end  

end  

return $RU$  

**Algorithm 1**: Unexpected-AR mining algorithm.

---

2The learning of the probabilities/parameters of BNs is a well studied topic, see (Heckerman, 1998; Korb and Nicholson, 2003) for example.
Similarly, the definition of unexpectedness for dependence rules (introduced in Appendix B) is defined on the difference between the interest of a rule and its inferred interest. The interest of dependence rules is defined in Equation B.1.

**Definition 5** (Unexpectedness for Dependence Rules). Given a Bayesian network $BN$ and data $D$, the unexpectedness of a dependence rule $r = \{A, B\}$ is defined as:

$$U(r) = \left| I(r) - \hat{I}_{BN}(r) \right|, \quad (5.2)$$

where $I$ is the interest of the rule, and $\hat{I}_{BN}$ is the interest inferred from $BN$ calculated by:

$$\hat{I}_{BN}(r) = \frac{\hat{P}_{BN}(A, B)}{P(A)P(B)}$$

The algorithm of mining unexpected dependence rules is shown in Algorithm 2.

**Algorithm 2**: Unexpected-DR mining algorithm.

```
Input: D: data, BN_{local}: local Bayesian network, R: dependence rules, 
\epsilon_U: threshold for unexpectedness
Result: R_U: Unexpected rules
R_U \leftarrow \emptyset;
forall r \in R do
    BN \leftarrow BN_{local}(r) \text{ (by Algorithm 4)};
    LearnParameters(BN, D);
    U(r) = \left| I(r) - \hat{I}_{BN}(r) \right|;
    if U(r) > \epsilon_U then
        Append r to $R_U$;
    end
end
return $R_U$
```

The algorithms above iteratively evaluate each association/dependence rule by constructing a local BN, inferring the probability from the BN, and calculating the difference between the actual probability of the rule and the inferred one; if the difference is greater than a threshold, the rule is appended to the set of unexpected rules.

**Run Time Test**
We test the run time of Algorithm 1 using our implementation (reported in Appendix D) on a 2GHz CPU computer. The setting of rule mining is the same as the one used in the mining of ANZDATA (reported in Section 7.1). Figure 5.1 plots the run time of mining 100 to 1,000 rules (the curve of circle markers); it takes 257.3 seconds to process 1,000 rules; in other words, the implementation can process up to approximately 13,990 rules per hour.

The run time of searching local BN subroutine (will be discussed in Subsection 5.3.4) is also plotted in Figure 5.1 as the curve of square markers. The $d_{\text{max}}$ parameter of Algorithm 4 was set to 5.

![Figure 5.1: Run time test of Algorithm 1 and 4 in mining the ANZDATA. Curve of circle markers denotes the run time of Algorithm 1. Curve of square markers denotes the run time of Algorithm 4 (in Subsection 5.3.4).](image)

**5.3.2 Association Rules and Correlations**

Association rules, as their name suggest, are representing associations rather than correlations. From the statistical point of view, association rules sometimes are statistically independent. Please see the Problem 2 in Appendix B for example. In our iterative unexpectedness design, unexpected rules should trig-
5.3. PROPOSED DESIGN

An intuitive idea to resolve this problem is to use the test for independence in classical statistics. In the context of data mining, the test for independence for associations has been investigated in (Bay and Pazzani, 1999; Bayardo and Agrawal, 1999; Brin et al., 1997; Silverstein et al., 1998) and etc. Specifically, Silverstein et al. (1998) defined dependence rules which are a kind of generalized association rules that identify statistical dependence in itemsets. The significance of dependence of rules is measured via the chi-squared test ($\chi^2$ test) for independence from classical statistics. Therefore, for ensuring the presented association rules are not presenting independent relationships, we could apply the $\chi^2$ test to mined association rules [The $\chi^2$ test is briefly introduced in Section B.1].

In the proposed approach, association rules are mined first, and then filtered by the test of independence; therefore, the filtered association rules have the context of dependence rules.

5.3.3 Acquiring User’s Knowledge

The ‘knowledge acquisition bottleneck’ is a general problem in knowledge engineering: acquiring more knowledge means more overhead. In the case of unexpected pattern mining, a user’s expectation has to be acquired somehow. In terms of Bayesian networks, there are two kinds of knowledge that need to be acquired: the structure and the parameters of a Bayesian network. However, it is impractical to design an approach that requires a user to manually assign Bayesian network’s parameters due to the exponential sizes of conditional probability tables. In the design, the only format of expectation we ask a user to provide is their knowledge about associations. The task is designed as follows. For each variable in data, ask a user to assign other variables that he/she thinks are relates-to the given variable. For example, Table 5.1 shows a sample of some associations of variables given by the expert for the ANZDATA. The acquired table of associations is equivalent to an association graph (AG).

**Problem 1** (Cyclic Association Graph). *Because AG is manually assigned, it inevitably contains cycles that are not allowed in Bayesian networks.*
For converting AG into DAG, we propose a simple flipping algorithm. Let us first consider the simplest case:

\[ \begin{array}{c}
A \\
\rightarrow
\end{array} \quad \begin{array}{c}
B
\end{array} \]

In the case \( A, B \) being bi-directed, eliminating one edge would make the graph acyclic; or, equivalently, one edge is flipped into another orientation. Let us consider another case for three vertices graphs:

\[ \begin{array}{c}
A \\
\rightarrow
\end{array} \quad \begin{array}{c}
B \\
\rightarrow
\end{array} \quad \begin{array}{c}
C
\end{array} \]

In the case \( A, B, C \) forming a cycle, flipping any one of the edges would make the graph acyclic; for instance:

\[ \begin{array}{c}
A \\
\rightarrow
\end{array} \quad \begin{array}{c}
B \\
\rightarrow
\end{array} \quad \begin{array}{c}
C
\end{array} \]

\[ \begin{array}{c}
A \\
\rightarrow
\end{array} \quad \begin{array}{c}
B \\
\rightarrow
\end{array} \quad \begin{array}{c}
C
\end{array} \]

Intuitively, for a cyclic path in a graph, flipping any edge would break the cycle. Now we consider a whole graph with many cycles. If we randomly flip an edge of a cycle, the graph would remain cyclic because there are still other cycles and the flipped edge might incur new cycles; however, if we iteratively apply this strategy to the graph until no more cycles, then the graph becomes acyclic.

Before presenting the algorithm of this approach, let us discuss an issue regarding changing the context encoded in the original graph. For the basic unit \( A \rightarrow B \) means \( A \text{ relates-to } B \), under our definition of an arrow. Therefore, \( B \rightarrow A \) is equivalent to \( A \rightarrow B \) because \text{relates-to} is a symmetric descriptor. Conclusively, the converted DAG has the same message as the original AG.

For simplifying the design of knowledge acquisition, this argument overlooks causality or other advanced issues, e.g. d-separation, in Bayesian networks.
5.3. PROPOSED DESIGN

(e.g. Pearl (1998); Korb and Nicholson (2003); Boneh et al. (2006)).

We now present a simple algorithm for converting AG to DAG (Algorithm 3). The algorithm iteratively and randomly flips an edge of every cycle until done. There is a FindCyclicPath(G) function that returns the first cyclic path found in graph G. The RandomPickEdge(path) function returns an edge that is randomly picked from the edges in path. The flipping function FlipEdge(G,e) changes the orientation of e in G. The graphs, AG and DAG, are in matrix format.

Input: AG: Association Graph
Result: DAG: Directed acyclic graph

\[\text{DAG } \leftarrow \text{AG};\]
\[\text{stop } \leftarrow \text{False};\]

\[\text{while } \text{stop } = \text{False} \text{ do}\]
\[\quad \text{path } = \text{FindCyclicPath}(\text{DAG});\]
\[\quad \text{if } \text{path} = \emptyset \text{ then}\]
\[\quad \quad \text{stop } \leftarrow \text{True}\]
\[\quad \text{else}\]
\[\quad \quad \text{edge } = \text{RandomPickEdge}(\text{path});\]
\[\quad \quad \text{DAG } \leftarrow \text{FlipEdge}(\text{DAG}, \text{edge});\]
\[\text{end}\]

\[\text{return } \text{DAG}\]

Algorithm 3: AG-to-DAG converting algorithm.

5.3.4 Local Bayesian Networks for Rules

In practice, the exponential growing CPTs of Bayesian networks restrict the size of parent nodes of a child node, due to the limited memory of computers. For example, it is impossible to build a full Bayesian network in the case of mining ANZDATA because the degree of vertices would be over 30. Instead of using full KB (the acquired DAG), we propose that a partial KB may be used for each rule alternatively. In other words, the idea is to build a local Bayesian network (local BN) for each rule.

For an association rule \(A \rightarrow B\), what we need to know from the KB is the expectation of \(P(B|A)^3\). Inferring \(P(B|A)\) from a Bayesian network involves

\[3\text{Based on the definition of unexpectedness in Equation 5.1}\]
three possible pathways: (1) paths from $A$ to $B$ and vice versa, (2) common causes of $A, B$ and (3) their common effects. Therefore, the local BN only needs to contain nodes and edges that cover the possible pathways. Additionally, because the only evidence is the left-hand-side of the association rule, i.e. $A$, any path between $A$ and $B$ provides sufficient information for inference. Consider the example below:

![Diagram](image)

Either $A \rightarrow X \rightarrow B$, $A \rightarrow Y_1 \rightarrow Y_2 \rightarrow Y_3 \rightarrow B$ or $A \rightarrow Z_1 \rightarrow \cdots \rightarrow B$ provides a pathway for inferring $P(B|A)$. This means we can further reduce the local BN by removing nodes of longer pathways, e.g. $Y, Z$ in the example.

We present a simple algorithm for generating a local BN based on a given association rule. Let $AR$ be the association rule having one LHS\(^4\) attribute or two LHS attributes: $A \rightarrow B$ or $A_1, A_2 \rightarrow B$. $DAG$ be the structure of the Bayesian network converted by Algorithm 3. The $FindAllPaths(G_{ud}, A_i, B, d_{max})$ function searches all possible paths between $A$ and $B$ within depth $d_{max}$ and returns found paths. Note that the parameters of the generated $BN_{local}$ are not

\(^4\)Left hand side.
5.3. PROPOSED DESIGN

estimated yet; they will be learned in Algorithm 1.

**Input:** \( AR \): association rule; \( DAG \): directed acyclic graph; \( d_{\text{max}} \): maximum depth for search.

**Result:** \( BN_{\text{local}} \): local Bayesian network

\[
G_{ud} \leftarrow OR(DAG, DAG^T) \quad (G_{ud} \text{ is a undirectional graph});
\]

\( A \leftarrow AR\text{'s LHS}; \)

\( B \leftarrow AR\text{'s RHS}; \)

\( \text{nodes} \leftarrow \emptyset; \)

for all \( A_i \in A \) do

\[
\text{paths} = \text{FindAllPaths}(G_{ud}, A_i, B, d_{\text{max}});
\]

\[
v = \bigcup_{\text{vertices} \in \text{paths}} \text{vertices};
\]

\( \text{nodes} = \text{nodes} \cup v; \)

end

\( \text{nodes} = \text{nodes} \cup \{A, B\}; \)

\( BN_{\text{local}} \leftarrow \text{DAG}(\text{nodes}); \)

return \( BN_{\text{local}} \)

Algorithm 4: Local-BN generating algorithm.

---

5.3.5 Computability of Unknown

Recall the nomenclature of unexpectedness (Figure 2.2), ‘unknown’ is a purely subjective term and ‘unrecognised’ is its computational counterpart. Taking rule-based representation as an example, if a KB is empty, then any rule is unrecognised. However, other knowledge representations cannot assess the quality of ‘unrecognised’, instead they can only assess ‘contradictory’.

The Bayesian network based representation is incapable of assessing ‘unrecognised’ because of its semantic of arcs. Generally, an arc between nodes \( A, B \) represents that \( A \) and \( B \) are dependent, otherwise independent. In practice, if a user leaves \( A \) and \( B \) separated (e.g. Figure 5.2), either the user thinks \( A \) and \( B \) are independent or the user does not know their relation. Therefore, we could not say the rule \( A \rightarrow B \) is unrecognised due to the absence of connection between \( A \) and \( B \) in the KB. On the other hand, we can determine whether the rule contradicts the KB via the unexpectedness measure.
5.3.6 Overall Design of Unexpected Rule Mining

We have discussed the rationale for utilising Bayesian networks as the knowledge representation platform for unexpected rule mining in (Chapter 2.4). We have also discussed the fact that association rules do not represent correlations, which is essential in the context of Bayesian networks, and we propose using a statistical test to filter out independent rules from association rules (Subsection B.1). Thirdly, the method and practical issues of knowledge acquisition are discussed in Subsection 5.3.3.

Because a user’s knowledge is dynamic under the interaction with mined rules, the procedure of unexpected rule mining is designed in an iterative manner so that the user could add new learned knowledge to KB and discover another set of unexpected rules. The overall design is plotted in Figure 5.3. Firstly, dependent association rules are mined via $\chi^2$ test on association rules. On the side of knowledge acquisition, a user inputs his/her knowledge in the format of association graph. Secondly, local Bayesian networks are generated for every dependent AR by the method of KB preprocessing described in Subsection 5.3.3. Then, unexpected rules are mined by the algorithm proposed in Subsection 5.3.1.

At the next iteration of unexpected rules mining, a user could update the KB by refining the association graph based on the mined unexpected rules; then repeat the same procedure again to get another set of unexpected rules. In terms of termination criterion of the iterative procedure, we propose that a user could stop the process when the maximal unexpectedness (Equation 5.1 or 5.2) is lower than a threshold.

**Interactive Platform**

Because the design of unexpectedness is composed of elements for pattern interpretation and knowledge acquisition, there is a need for a platform for...
Here we report a case study of the proposed design to demonstrate how it works, and compare it with the design of (Jaroszewicz and Simovici, 2004). The case study uses the KSL dataset and is conducted in association rules mining.

### 5.4.1 Danish Geriatric (KSL)

In the first case study, we compare the proposed method with the method by Jaroszewicz and Simovici (2004) which inspires our work. We use the same

---

5The function for explanations will be discussed in next chapter.
dataset as used in the experiment of (Jaroszewicz and Simovici, 2004) - the KSL dataset which comes from a study measuring health and social characteristics of representative samples of Danish geriatrics, taken in 1967 and 1984 (more details are in Chapter 3).

The proposed method of unexpectedness is inspired by the work of (Jaroszewicz and Simovici, 2004), and we modify and add some elements to resolve potential problems. First, we use confidence (conditional probability) rather than support (joint probability) in the definition of unexpectedness because of the users’ interest is in conditional probability. Of course, any other objective interestingness measure could be adapted easily as long as the Bayesian network is capable of inferring the probabilities for the measure. The second change is the test for statistical independence for association rules; because the rules are used to guide the update of Bayesian networks. Thirdly, because the user assigned graph is usually cyclic which is forbidden in Bayesian networks, we have devised a simple method to convert the cyclic graph to an acyclic one. Finally, in addressing the problem of computation efficiency, we propose to use a local Bayesian network for a given rule to speed up the computation. Table 5.2 makes a comparison between the proposed method with the approach of (Jaroszewicz and Simovici, 2004).

First Iteration

In the experiment, the initial KB of both approaches are set to the example provided by Jaroszewicz and Simovici (2004), based on their (non-expert) knowledge; it is plotted in Figure 5.4. Since the input is already a DAG, we do not need the conversion of cyclic graphs to DAGs here (steps 2 and 8 in Table 5.2).

With the thresholds of support = 0.02 and confidence = 0.7, there are 432 association rules mined; among the association rules, there are 113 rules to be statistically dependent. The top 5 unexpected rules, mined in Step 5, are listed in Table 5.3.

In comparison, the method of (Jaroszewicz and Simovici, 2004) reports two interesting itemsets (listed in Table 5.4). The first itemset \{FEV,Sex\} appears as the second unexpected rule in our approach: \( \text{FEV}=264.7 \sim 378 \rightarrow \text{Sex}=\text{male} \) (\text{supp} = 0.08, \text{conf} = 0.98). For the second itemset \{Alc,Year\}, it is also found as unexpected in Rules 1, 3 and 5.
5.4. CASE STUDY

<table>
<thead>
<tr>
<th>Step</th>
<th>Our approach</th>
<th>Method of (Jaroszewicz and Simovici, 2004)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>User assign the AG (association graph).</td>
<td>User construct the Bayesian network.</td>
</tr>
<tr>
<td>2</td>
<td>Convert AG to DAG.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Independence test for association rules.</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Mine unexpected rules.</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>User interpret results.</td>
<td>User interpret results.</td>
</tr>
<tr>
<td>7</td>
<td>User update AG.</td>
<td>User update the Bayesian network.</td>
</tr>
<tr>
<td>8</td>
<td>Convert AG to DAG.</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>User interpret results.</td>
<td>User interpret results.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(n)</td>
<td>Until no significant unexpected rules mined.</td>
<td>Until no significant unexpected frequent itemsets mined.</td>
</tr>
</tbody>
</table>

Table 5.2: Comparison of the proposed unexpectedness design and the method by Jaroszewicz and Simovici (2004).

Figure 5.4: The initial Bayesian network as background knowledge, from the experiment of (Jaroszewicz and Simovici, 2004).
### Chapter 5. Unexpected Pattern Mining

<table>
<thead>
<tr>
<th>No.</th>
<th>Rule</th>
<th>supp</th>
<th>conf</th>
<th>U</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\text{Kol}=992-1344, \text{Alc}=\text{seldom} \rightarrow \text{Year}=1967$</td>
<td>0.02</td>
<td>0.78</td>
<td>0.48</td>
</tr>
<tr>
<td>2</td>
<td>$\text{FEV}=264.7-378 \rightarrow \text{Sex}=\text{male}$</td>
<td>0.08</td>
<td>0.98</td>
<td>0.45</td>
</tr>
<tr>
<td>3</td>
<td>$\text{FEV}=38 \sim 151.3, \text{Year}=1967 \rightarrow \text{Alc}=\text{seldom}$</td>
<td>0.13</td>
<td>0.88</td>
<td>0.39</td>
</tr>
<tr>
<td>4</td>
<td>$\text{Kol}=992 \sim 1344 \rightarrow \text{Sex}=\text{female}$</td>
<td>0.03</td>
<td>0.86</td>
<td>0.38</td>
</tr>
<tr>
<td>5</td>
<td>$\text{Smok}=\text{no}, \text{Year}=1967 \rightarrow \text{Alc}=\text{seldom}$</td>
<td>0.09</td>
<td>0.86</td>
<td>0.36</td>
</tr>
</tbody>
</table>

Table 5.3: Unexpected rules of proposed method in the first iteration.

<table>
<thead>
<tr>
<th>Itemset</th>
<th>Interestingness</th>
</tr>
</thead>
<tbody>
<tr>
<td>{FEV,Sex}</td>
<td>0.0812</td>
</tr>
<tr>
<td>{Alc,Year}</td>
<td>0.0810</td>
</tr>
</tbody>
</table>

Table 5.4: Interesting/unexpected itemsets of first iteration reported in (Jaroszewicz and Simovici, 2004).

Now since the two methods generate different unexpected rules and itemsets, the modification of the network structures is therefore diverged. Based on the five unexpected rules, we add six links: $\text{Year} \rightarrow \text{Kol}, \text{Year} \rightarrow \text{Alc}, \text{Sex} \rightarrow \text{FEV}, \text{FEV} \rightarrow \text{Alc}, \text{Sex} \rightarrow \text{Kol}, \text{Smok} \rightarrow \text{Alc}$; Figure 5.5 shows the new links in dot lines. On the other hand, Jaroszewicz and Simovici (2004) update the Bayesian network with $\text{Sex} \rightarrow \text{FEV}$ and $\text{Year} \rightarrow \text{Alc}$ (Figure 5.6).

**Second and Third Iteration**

In the second iteration, new unexpected rules are mined based on the modified KB; Table 5.5 shows the result. According to the rules, we decide to add the

![Figure 5.5: First modification of BN by our approach.](image)

![Figure 5.6: First modification of BN by Jaroszewicz and Simovici (2004).](image)
5.4. CASE STUDY

following edges: Year $\rightarrow$ Smok, Alc $\rightarrow$ Kol, Year $\rightarrow$ FEV (Figure 5.7).

In comparison, the method of (Jaroszewicz and Simovici, 2004) finds one interesting itemset in the second iteration (Table 5.6). Consequently, Jaroszewicz and Simovici (2004) added two edges: Year $\rightarrow$ Kol and Sex $\rightarrow$ Kol to the network structure (Figure 5.8).

<table>
<thead>
<tr>
<th>No.</th>
<th>Rule</th>
<th>$\text{supp}$</th>
<th>$\text{conf}$</th>
<th>$U$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Kol=992-1344 $\rightarrow$ Alc=seldom</td>
<td>0.03</td>
<td>0.79</td>
<td>0.19</td>
</tr>
<tr>
<td>2</td>
<td>FEV=264.7-378, Smok=no $\rightarrow$ Year=1984</td>
<td>0.02</td>
<td>0.88</td>
<td>0.18</td>
</tr>
<tr>
<td>3</td>
<td>FEV=264.7-378, Year=1967 $\rightarrow$ Smok=yes</td>
<td>0.02</td>
<td>0.88</td>
<td>0.16</td>
</tr>
<tr>
<td>4</td>
<td>Smok=no, Sex=male $\rightarrow$ Year=1984</td>
<td>0.06</td>
<td>0.81</td>
<td>0.10</td>
</tr>
<tr>
<td>5</td>
<td>FEV=264.7-378, Work=no $\rightarrow$ Year=1984</td>
<td>0.06</td>
<td>0.83</td>
<td>0.09</td>
</tr>
</tbody>
</table>

Table 5.5: Unexpected rules of proposed method in the second iteration.

<table>
<thead>
<tr>
<th>Itemset</th>
<th>Interestingness</th>
</tr>
</thead>
<tbody>
<tr>
<td>{Kol,Sex,Year}</td>
<td>0.0665</td>
</tr>
</tbody>
</table>

Table 5.6: Interesting/unexpected itemsets of second iteration reported in (Jaroszewicz and Simovici, 2004).

Figure 5.7: Second modification of BN by our approach.

Based on the modified background knowledge, the third iteration mines a new set of unexpected rules (shown in Table 5.7). In comparison, the method of (Jaroszewicz and Simovici, 2004) finds five interesting itemsets with a technique called topological pruning in the third iteration (Table 5.8).
Table 5.7: Unexpected rules from the proposed method in the third iteration.

<table>
<thead>
<tr>
<th>No.</th>
<th>Rule</th>
<th>supp</th>
<th>conf</th>
<th>U</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>FEV=264.7-378, Year=1984 → Work=no</td>
<td>0.06</td>
<td>0.77</td>
<td>0.08</td>
</tr>
<tr>
<td>2</td>
<td>FEV=38-151.3, BMI=26.2-36.5 → Hyp=yes</td>
<td>0.12</td>
<td>0.75</td>
<td>0.08</td>
</tr>
<tr>
<td>3</td>
<td>FEV=264.7-378 → BMI=15.8-26.2</td>
<td>0.08</td>
<td>0.71</td>
<td>0.07</td>
</tr>
<tr>
<td>4</td>
<td>FEV=264.7-378, Work=no → Year=1984</td>
<td>0.06</td>
<td>0.83</td>
<td>0.06</td>
</tr>
<tr>
<td>5</td>
<td>FEV=38-151.3, Sex=male → Work=no</td>
<td>0.10</td>
<td>0.85</td>
<td>0.06</td>
</tr>
</tbody>
</table>

Table 5.8: Interesting/unexpected itemsets of third iteration with topological pruning reported in (Jaroszewicz and Simovici, 2004).

<table>
<thead>
<tr>
<th>Itemset</th>
<th>Interestingness</th>
</tr>
</thead>
<tbody>
<tr>
<td>{FEV,Alc,Year}</td>
<td>0.0286</td>
</tr>
<tr>
<td>{Kol,BMI}</td>
<td>0.0144</td>
</tr>
<tr>
<td>{Kol,Alc}</td>
<td>0.0126</td>
</tr>
<tr>
<td>{Smok,Sex,Year}</td>
<td>0.0121</td>
</tr>
<tr>
<td>{Alc,Work}</td>
<td>0.0110</td>
</tr>
</tbody>
</table>

To summarise, the case study demonstrates that the proposed design can discover unexpected rules efficiently: within just two iterations and 10 rules to propagate the KB, the highest unexpectedness drops from 0.48 to 0.08, a 600% improvement. In comparison with (Jaroszewicz and Simovici, 2004), the proposed design propagates KB in a faster pace, and unexpected rules were mined in earlier iterations. Figure 5.9 shows a comparison that some of the unexpected itemsets of third iteration (Table 5.8) are already discovered in the first two iterations of the proposed approach.

5.5 Chapter Summary

In this chapter, we developed new building blocks for unexpectedness using BN based knowledge representation. They are: statistical independence test for association rules, association graphs for knowledge acquisition and an algorithm for converting AG to DAG; and exploiting local BN to address the problem of scalability in CPTs for datasets with a large number of attributes. The independence test is introduced for ensuring the presented association rules are dependent so that updating of the BN-based KB is meaningful. The proposed association graph could potentially alleviate the difficulty of asking
the user to assign a DAG for constructing BN.

The design for unexpectedness maintains the KB in the manner of acquire-apply-update procedure; unexpected rules are mined iteratively according to the updated KB; to the user, the reward of providing knowledge is a succinct set of unexpected rules.

Figure 5.9: Earlier discovery of unexpected rules compared to unexpected frequent itemsets.

<table>
<thead>
<tr>
<th>First iteration</th>
<th>Second iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1  Kol=992-1344, Alc=seldom → Year=1967</td>
<td>1  Kol=992-1344 → Alc=seldom</td>
</tr>
<tr>
<td>2  FEV=264.7-378 → Sex=male</td>
<td>2  FEV=264.7-378, Smok=no → Year=1984</td>
</tr>
<tr>
<td>3  FEV=38~151.3, Year=1967 → Alc=seldom</td>
<td>3  FEV=264.7-378, Year=1967 → Smok=yes</td>
</tr>
<tr>
<td>4  Kol=992~1344 → Sex=female</td>
<td>4  Smok=no, Sex=male → Year=1984</td>
</tr>
<tr>
<td>5  Smok=no, Year=1967 → Alc=seldom</td>
<td>5  FEV=264.7-378, Work=no → Year=1984</td>
</tr>
</tbody>
</table>
Chapter 6

Explanation Generation

In the course of our investigation into discovering surprising rules, we identified an important characteristic of user interaction. Derived from our observation that during the presentation of mined rules, the user (also an expert) interpreted each mined rule in the context of explaining it, it seems that finding explanations plays a critical role in pattern interpretation. Therefore, finding explanations is important to the cycle of mining unexpected patterns (as Figure 2.4 depicted).

Existing data mining technology generally leaves the task of pattern interpretation to KDD users, who are usually experts with a great deal of domain knowledge. Although users eventually have to interpret rules in their own way, we propose that computational assistance for pattern interpretation may be extremely beneficial.

During the course of the research, we encountered situations in which the user (a nephrologist) gave an explanation to a rule, then wondered whether the explanation was consistent with the facts in the data, because sometimes the facts in the data do not support the expert’s medical knowledge. Existing data mining techniques do not necessarily provide means for validating explanations in data (Fayyad et al., 1996; Chen and Liu, 2005); thus the task of validation needs to be done manually.

Towards the aim of computational assistance for pattern interpretation, we developed a solution not only for explanation validation but also for explanation generation (Kuo et al., 2008). The method for explanation generation is based on the context of probabilistic dependencies and is modeled in reasoning chains by means of comparing the inferred conditional probability with the actual conditional probability of a rule. The reasoning chains of explanations are modeled in Bayesian networks in order to be generalised to more complex
6.1 Motivation

In presenting (computationally) unexpected rules to a user, the user typically tries to reason about these rules by explaining them; yet, the reasoning task is difficult since these rules might be surprising to the user. Sometimes, rules are regarded as surprising at first glance, then become explicable in retrospect if a user comes up with some related factors; yet, a user might still wonder whether his/her explanation is supported by the data. On the other hand, if a user cannot immediately come up with any reasonable explanation for a rule, they might wonder whether there are some possible explanations contained within the data. In addressing this problem, the chapter presents a novel idea to computationally generate explanations from data.

6.2 Definition

The idea of the format of an explanation comes from our experience in a case study of medical data when discussing the result with a domain expert (Kuo et al., 2007). The medical dataset is the ANZDATA described in Section 3.1.1.

We illustrate our idea using a mined association rule stating that when kidney failure patients are not diagnosed with cancer, they are more likely to die of cardiovascular disease:

\[ \neg \text{Cancer} \rightarrow \text{CardiovascularDeath} \quad (6.1) \]

The medical expert interpreted this rule by giving an explanation: ‘noting that when patients have cancer, they usually die of cancer; in other words, when patients are not diagnosed with cancer they surely do not die of cancer and they are likely to die of cardiovascular disease.’ We can formulate the explanation as:

\[ \neg \text{Cancer} \rightarrow \neg \text{DieOfCancer} \rightarrow \text{CardiovascularDeath} \quad (6.2) \]

While we can verify the association in the data, we can also find the data supports the explanation in terms of conditional probability. The conditional
relative frequency in the data is:

$$P(\text{CardiovascularDeath}|\neg\text{Cancer}) = 0.557.$$  \hfill (6.3)

If we apply the chain rule provided by (Korb and Nicholson, 2003, chap. 1) to infer the probability of \text{CardiovascularDeath} by setting \text{DieOfCancer} as the intermediate variable, we can get a similar conditional probability:

$$\hat{P}(\text{CardiovascularDeath}|\neg\text{Cancer}) = P(\text{CardiovascularDeath}|\text{DieOfCancer}) \cdot P(\text{DieOfCancer}|\neg\text{Cancer}) + P(\text{CardiovascularDeath}|\neg\text{DieOfCancer}) \cdot P(\neg\text{DieOfCancer}|\neg\text{Cancer}) = 0 \cdot 0.714 + 0.538 \cdot 1 = 0.538.$$  \hfill (6.4)

Therefore, the probabilistic dependency in the data is consistent with the expert’s explanation. Now we give a general definition of an explanation for an association rule in terms of probability.

**Definition 6. \( \varepsilon \)-Explanations**

Given an association rule \( A \rightarrow B \) with confidence \( = P(B|A) \), we say a variable \( X \) is a factor (or an intermediate variable/intermediate) in an explanation if \( X \) can approximate the confidence of the rule via the chain rule in probability:

$$|P(B|A) - \hat{P}(B|A)| < \varepsilon,$$

where \( \hat{P}(B|A) = \sum_X P(B|X)P(X|A) \).

The explanation is written \( A \rightarrow X \rightarrow B \).

Def. 6 models reasoning chains by means of comparing the inferred conditional probability with the actual conditional probability of a rule.

**Example 2** (Inferred Conditional Probabilities of Different Intermediates).

This example shows the difference of inferred conditional probabilities when substituting different variables as the intermediates for an explanation. The ANZDATA introduced in Subsection 3.1.1 is used in this example.

Figure 6.1 shows the result for the rule \( \neg\text{Cancer} \rightarrow \text{CardiovascularDeath} \) which was discussed above; the \text{DieOfCancer} (a synthesized variable) appears
to be the one which can most precisely approximate the conditional probability of the rule, while there is a noticeable gap between \texttt{DieOfCancer} and other variables. The result confirms the expert's explanation of this rule which was calculated in Equation 6.4.

\begin{figure}[h]
\centering
\includegraphics[width=0.7\textwidth]{figure6_1.png}
\caption{Inferred conditional probabilities of different intermediates, $X$, for the explanation $\neg$Cancer $\rightarrow X \rightarrow$ CardiovascularDeath.}
\end{figure}

\textit{Let us examine another association rule – Diabetes $\rightarrow$ CardiovascularDeath – which has confidence $= P(\text{CardiovascularDeath}|\text{Diabetes}) = 0.5508$. The Diabetes variable includes Type 1 and Type 2 diabetes. In Figure 6.2, diabetes2 (Type 2 diabetes) appears to have the most close inferred conditional probability to the rule’s confidence. The result is reasonable because diabetes2 is actually part of the Diabetes variable.}
Figure 6.2: Inferred conditional probabilities of different intermediates, X, for the explanation \textit{Diabetes} \(\rightarrow\) \(X\) \(\rightarrow\) \textit{CardiovascularDeath}.

### 6.2.1 Modeling Explanations in Bayesian Networks

Def. 6 deals with only three variables: \(A, B\) and \(X\). However, association rules sometimes have more than one variable on LHS. Furthermore, there is no reason to limit explanations to only one factor at a time. For more complex explanations, we introduce the use of Bayesian networks for modeling explanations. For example, Figure 6.3 depicts an imaginary network structure that could be an explanation; in reality, searching such complex network structures is as hard as structure learning in Bayesian networks (Verma and Pearl, 1990a; Cooper and Herskovits, 1992; Spirtes et al., 2000; Pearl, 2000; Korb and Nicholson, 2003; de Campos, 2006).

The task of modelling explanations in BNs is similar to earlier work of causal diagrams and structure learning in BNs, which will be discussed in Section 6.4. However, they are still different in terms of ‘conditional independence’, observational/experimental data, causality, search space and scoring metrics. Therefore, the task of modelling explanations is a different problem to previous research. Although the heuristics of structure learning of BNs in earlier work could be used for speeding up the search of complex ‘explana-
6.2. DEFINITION

Figure 6.3: An imaginary example of a complex explanation. The intermediates, $X$, could be a network structure.

In the early phase of idea formalisation, we tend to keep the size of explanations manageable and its algorithm simple.

Let $\mathcal{A} = \{A_1, A_2, ..., A_m\}$ be a set of variables (attributes) at the LHS of a rule and $B$ be a variable (attribute) at the RHS of the rule; $\mathcal{X} = \{X_1, X_2, ..., X_r\}$ be a set of variables (attributes) serving as the factors in an explanation. The definition of a Bayesian networks as an explanation is given below.

**Definition 7. Bayesian Networks as $\varepsilon$-Explanations**

Given an association rule $\mathcal{A} \rightarrow B$ with confidence $= P(B|\mathcal{A})$, a Bayesian network $G$ consists of nodes $\{\mathcal{A}, B, \mathcal{X}\}$ is a $\varepsilon$-explanation if it satisfies the following requirements.

1. $\mathcal{X} \cap (\mathcal{A} \cup B) = \emptyset$. Every $X_i$ in $\mathcal{X}$ is not in $\mathcal{A}$ or $B$.
2. There is no direct connection between nodes in $\mathcal{A}$ and node $B$. (Assuming $\mathcal{A}$ and $B$ are indirectly dependent.)
3. Every node $a_i$ in $\mathcal{A}$ connects to one or more nodes in $\mathcal{X}$. There are no arcs among the nodes of $\mathcal{A}$.
4. There are no isolated nodes in $\mathcal{X}$. Each node $X_i$ has degree at least two.
5. The inferred conditional probability $\tilde{P}_G(B|\mathcal{A})$ is within $\varepsilon$ of the conditional relative frequency in the data:

$$|P(B|\mathcal{A}) - \tilde{P}_G(B|\mathcal{A})| < \varepsilon$$  (6.6)

$^1$In the thesis, we consider only one variable on the right-hand-side.
Therefore, the problem of finding explanations is reduced to finding a set of intermediate variables \( \mathcal{X} = \{X_1, X_2, \ldots, X_r\} \) and a Bayesian network \( G \) that can approximate the conditional relative frequency \( P(B|A) \) when \( a_1, a_2, \ldots, a_m \in \mathcal{A} \) is known. Figure 6.4 shows a few examples of the skeletons of Bayesian networks as possible explanations; the orientation of arcs can be assigned freely as long as the network \( G \) remains acyclic. We will further discuss statistically equivalent (Markov equivalent) models in the section of discussion (Sec. 6.3.1) showing that some graphs with same skeleton but different arc orientations are statistically equivalent.

In terms of dependence rules \( \{A, B\} \), we can treat them in the same way as association rules that \( P(B|A) \) or \( P(A|B) \) are the target of explanations. One might question that whether the context of conditional probability in explanations is suitable for dependence rules. Actually, the interest of dependence rules has the context of conditional probability:

\[
I_{\text{rule}}(ab) = \frac{P(ab)}{P(a)P(b)} = \frac{P(b|a)}{P(b)}.
\]

The question of the suitable size of explanations needs further research, however we offer a basis for limiting the number of variables involved by drawing on research in psychology. Recent research (Halford et al., 2005) has tested how many variables can human process and concluded that “... These findings suggest that a structure defined on four variables is at the limit of human processing capacity.” In the early development of this approach, we have chosen to deal at most two variables in the LHS of a rule and two intermediate factors/variables at a time.

### 6.2.2 Graphical Semantics of Bayesian Networks

In previous subsection, we discussed modeling explanations in Bayesian networks in the level of skeletons. We now consider the semantics of graphs in terms of dependence (Korb and Nicholson, 2003, chap. 2). Regrading a simplest skeleton of explanations:

\[
A \longrightarrow X \longrightarrow B.
\]

There are three types of conditional independence for different arc orientations for the simplest skeleton above. Please note that \( A \) and \( B \) are dependent be-
Figure 6.4: Examples of the skeletons of Bayesian networks as explanations.

cause they are tested by independence test in the design of unexpectedness. The three types of conditional independence and their relations with explanations are discussed as following.

**Causal chains** A causal chain has consistent arc orientations between $A, X$ and $X, B$, i.e. $A \rightarrow X \rightarrow B$ or $A \leftarrow X \leftarrow B$. This type of graphs has the meaning that $A$ and $B$ are conditional independent under $X$, denoted as $A \perp B | X$ – knowing $A$ does not make any difference to our belief about $B$ if we already know $X$. In terms of explanations, a causal chain represents a reasoning chain: if $A$ then $X$, and if $X$ than $B$.

**Common causes** $A$ and $B$ have a common cause $X$; the graph looks like

$$A \leftarrow X \rightarrow B.$$
In this case, $A$ and $B$ are also *conditional independent* under $X$:

$$P(B|A \land X) = P(B|X) \equiv A \perp B | X.$$  

In terms of explanations, $A, B$ are dependent because of their cause $X$.

**Common effects** $X$ is the common effect of $A, B$; it is represented by a network $v$-structure: $A \rightarrow X \leftarrow B$. It has an opposite conditional independence structure to that of chains and common causes. That is, the parents $A, B$ are marginally independent ($A \perp B$), but become dependent given information about $X$ (i.e. $A, B$ are *conditional dependent*):

$$P(B|A \land X) \neq P(B|X) \equiv \neg(A \perp B | X).$$

In terms of explanations, the common effect structure represents that $A, B$ are marginally independent; however $A, B$ are actually statistically dependent. There is a conflict between the dependence of $A, B$ and the semantic of common effects; thus common effects should be ruled out from possible explanations.

Among the three types of graphs, we can see that *causal chains* and *common causes* are candidates of explanations, but *common effects* should be ruled out. The vital point of verifying $X$ in *causal chains* or *common causes* is the test of *conditional independence*. We can use *conditional mutual information* to test *conditional independence*. The conditional mutual information is defined as

$$I_{\text{mutual}}(A, B | X) = \sum_{a \in A} \sum_{b \in B} \sum_{x \in X} P(a, b, x) \log \frac{P(a, b | x)}{P(a | x)P(b | x)}.$$  \hspace{1cm} (6.7)

When $I_{\text{mutual}}(A, B | X)$ is close to zero, $A, B$ is conditional independent under $X$; in practice, a small threshold $\epsilon$ is chosen so that $I(A, B | X) < \epsilon$ means conditional independent. Based on the semantic of Bayesian networks, we augment the definition of explanations in terms of *conditional independence*.

**Definition 8. $\epsilon$-Explanations with Conditional Independence**

Following Def. 7, given a $\epsilon$-explanation $e$, we say that $e$ satisfies the conditional independence if:

- $A \perp B | X$. $A, B$ are conditional independent under $X$. 
6.3 Explanation-Generating Algorithm

In searching for relatively small explanations, five variables at most, we present an exhaustive search but simple algorithm for finding possible explanations for an association/dependence rule in Algorithm 5. Let $G$ be the set of all possible structures of Bayesian networks for explanations. The algorithm substitutes all possible combination of variables $X$ into a collection of possible structures (DAGs) of Bayesian networks $G$, then learns the conditional probability tables (CPTs) from data in the step $\text{LearnParameters}(\cdot, \cdot)$; finally the algorithm estimates whether the Bayesian network satisfies Equation 6.6.

**Input:** $D$: data, $G$: a collection of the structure of Bayesian networks, $r$: an association rule, $\varepsilon_E$: threshold for acceptance, $\text{Attr}$: all attributes of data.

**Result:** $E_r$: a set of explanations

$E_r \leftarrow \emptyset$;

$\mathcal{A} \leftarrow r.A$; $B \leftarrow r.B$: attributes of the rule;

$a \leftarrow r.a$; $b \leftarrow r.b$: values of variables;

forall $G \in G$ do

forall $(X \in \text{Attr}) \land (X \notin \mathcal{A} \cup B)$ do

Build a Bayesian network $G$ from $\mathcal{A}, X, B$;

LearnParameters($G, D$);

Compute $PD = P(B = b | A = a)$ from data;

Compute $\hat{P} = P_G(B = b | A = a)$ from $G$;

if $|PD - \hat{P}| < \varepsilon_E$ then

if $\mathcal{A} \perp B | X$ then

Append $G$ to $E_r$;

end

end

end

return $E_r$

**Algorithm 5**: Explanation-generating algorithm for a rule.

The concept of the algorithm is to exhaustively search for possible structures and intermediate variables which satisfy Def. 8. The search space of the intermediate variables $X$ is set to the attributes of data excluding the attributes of the input rule. The parameters (CPT) of constructed BNs are learned from data, so that the found ‘explanations’ are using the ‘facts’ in data, not necessarily being the same as the knowledge of users.
Analysis of Complexity
The time complexity of the algorithm for explanations is $O(n^2)$ where $n$ is the number of attributes in data under the circumstance that at most 2 additional factors are modeled. The complexity is analysed from the following reasons. If there is only one additional factor to be modeled, $|X| = 1$, the number of candidates for $X$ is exactly $n - |A| - |B|$; meanwhile the size of the templates for the structure of BN is a constant: $k_1 = |G_1|$; together the size of the search space is:

$$k_1(n - |A| - |B|).$$

If there are two additional factors to be modeled, $|X| = 2$, the size of the permutation and combination of candidates for $X$ is exactly $2\left(\frac{n - |A| - |B|}{2}\right)$; meanwhile the size of the templates for the structure of BN is still a constant: $k_2 = |G_2|$; together the size of the search space is:

$$k_2 \cdot 2\left(\frac{n - |A| - |B|}{2}\right) = k_2(n - |A| - |B|)(n - |A| - |B| - 1).$$

Therefore the upper bound of the complexity is $O(n^2)$ when $|X| = 2$. A generalization of the time complexity to $m$ additional factors would be $O(n^m)$ which is exponential growing in respect to the size of $X$. ■

6.3.1 Discussion
Association versus Causality
Humans seek explanations in the context of cause and effect. However, we have to clarify that the proposed method is only capable of finding explanations in the context of associations. There are two reasons for the limitation. First, the background knowledge of metadata is usually not modeled in general data mining practice; moreover, it is still a challenge for machines to reason causality based on the name of entities unless it is already modeled in the background knowledge. For example, variables named WeatherOutlook and TomorrowWeather seem meaningful to us; however they look just like $X$ and $Y$ to a computer. Furthermore, solving causality in this way will be a paradox because if we already know the relationship among variables, why do we need data mining?
6.3. EXPLANATION-GENERATING ALGORITHM

The second reason is that general data mining practice does not allow intervention which is one of the basic principles for identifying causation (Pearl, 2002). For example we can intervene the status of a switch to on/off repeatedly and see whether the light is on/off accordingly. However, to the best of data mining algorithms, it can only give rules such as IF light=on THEN switch=on or IF switch=on THEN light=on when the data is collected in a general tabular format.

Another matter of causality is in the representation of Bayesian networks. Although BNs are capable of modeling causal relationships, it does not ensure the orientations of arcs actually represent cause and effect. This property can be inspected from the Bayes' Theorem:

\[
P(B_k | A) = \frac{P(A | B_k) P(B_k)}{\sum_i P(A | B_i) P(B_i)}. \tag{6.8}
\]

For instance, by knowing the CPT of the left BN in below:

\[
\begin{array}{c}
A \\
\uparrow \uparrow \uparrow \uparrow \\
B
\end{array}
\quad
\begin{array}{c}
A \\
\uparrow \uparrow \uparrow \uparrow \\
B
\end{array}
\quad
\begin{array}{c}
A \\
\uparrow \uparrow \uparrow \uparrow \\
B
\end{array}
\quad
\begin{array}{c}
A \\
\uparrow \uparrow \uparrow \uparrow \\
B
\end{array}
\]

we can calculate the CPT of the right BN by Bayes' Theorem. Therefore the two BNs are equivalent in terms of probability in that the arc orientation is interchangeable.

For more complex Bayesian networks, we can determine whether two models are statistically equivalent by the theorem of Markov equivalent.

**Theorem 1. Markov Equivalent**

Any two causal models over the same variables that have the same skeleton and the same \(v\)-structure are Markov equivalent (Verma and Pearl, 1990b).

The \(v\)-structure represents the common effect on a variable, e.g. \(A \rightarrow X \leftarrow B\). Therefore, \(A \rightarrow X \rightarrow B\), \(A \leftarrow X \rightarrow B\) and \(A \leftarrow X \leftarrow B\) are statistically equivalent models and \(A \rightarrow X \leftarrow B\) is different from them (Korb and Nicholson (2003), p.164).

To sum up, the proposed method of finding explanations identifies associations among variables, not causation. The orientation of arcs should not and could not be perceived as cause-effect because of the reasons discussed.


Arcs Orientation in Presentation

Although the found explanation represents only associations, the user would still naturally interpret an arc/edge of two variables as cause and effect because of the arrow head. In the case of common cause in a graph

\[ A \leftarrow X \rightarrow B, \]

we find the user would be confused about how to reason the distribution of \( X \) given \( A, B \). Instead, we found that the reasoning chain \( (A \rightarrow X \rightarrow B) \) is a much more understandable format of explanations. Given the Markov Equivalent property, converting the orientation of arcs into reasoning chains does not change the statistical property of explanations. Therefore, in order to make the explanations more understandable, the explanations will be presented in the format of reasoning chains (the orientation of arcs will be adjusted) in the case studies.

Explanation’s Reasonableness and Rule’s Unexpectedness

Since the number of entities and relationships in a user’s knowledge is much greater than the number of attributes in the data, and the proposed method for explanations only explores the attributes in data, it implies that a user would usually have explanations for rules that include factors not observed in the data. Therefore, a user would not always find generated explanations to be reasonable; similarly, a user will not always classify unexpected rules as truly surprising. To help understand a user’s perspective of the data mining process we have proposed, we discuss four of the nine possible scenarios listed in Table 6.1. The ‘other’ situations happen when a user has different opinion than surprising/unsurprising or reasonable/unreasonable; we omit the discussion of them here but report some ‘other’ interpretations in our experiment.

The four cases (S-R, S-UR, U-R, U-UR) may be more clearly understood if we look at the ‘un/surprising’ status of a rule and ‘un/reasonable’ status of an explanation separately:

Surprising(to a user) When a rule is regarded as surprising to a user, it implies that the \( \text{KB} \) properly captures the user’s knowledge over the rule; that is, the user does not know the rule AND the \( \text{KB} \) does not contain the rule.
6.3. EXPLANATION-GENERATING ALGORITHM

<table>
<thead>
<tr>
<th>Rule</th>
<th>Reasonable Explanation</th>
<th>Unreasonable Explanation</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surprising (to a user)</td>
<td>S-R</td>
<td>S-UR</td>
<td>S-O</td>
</tr>
<tr>
<td>Unsurprising (to a user)</td>
<td>U-R</td>
<td>U-UR</td>
<td>U-O</td>
</tr>
<tr>
<td>Other</td>
<td>O-R</td>
<td>O-UR</td>
<td>O-O</td>
</tr>
</tbody>
</table>

Table 6.1: Possible interpretations from user about unexpected rule and its explanation. The acronyms are: S='Surprising', U='Unsurprising', R='Reasonable', UR='Unreasonable', and O='Other'.

Unsurprising (to a user) When a rule is regarded as unsurprising to a user, it implies that the \( KB \) has not fully captured the user’s knowledge over the rule, because the knowledge representation is limited, or the user finds it hard to articulate their knowledge completely; therefore, the acquired \( KB \) is usually incomplete.

Reasonable Explanation The associations in the explanation are consistent with the user’s knowledge.

Unreasonable Explanation There are multiple reasons for this situation. The first possibility is that the user thinks \( A \rightarrow B \) is directly related (direct relatedness); therefore, the explanation generation method would be redundant because it is designed to find additional factors. Secondly, if the factors for reasonable explanations are not in the observed data (hidden factors), the method cannot find them; instead, the method would find other factors in the observed data. The third possibility would be that the found factors, although computationally fit as explanations, are not in the user’s knowledge (unknown factors). In other words, the user does not know the associations in the explanation. This may happen because genuinely new knowledge has been generated (the best possible outcome). Finally, the distribution in the data could be different from user’s knowledge (different facts in data); therefore, the generated explanations would not fit the user’s knowledge, perhaps because they are dataset-specific relationships which are not representative of more general (but still domain specific) knowledge (or vice-versa); an example of this from the ANZDATA dataset, which records clinical measurements from dialysis patients only, would be that dialysis patients tend to survive better if overweight, which is not true of the general population.
CHAPTER 6. EXPLANATION GENERATION

<table>
<thead>
<tr>
<th>Observed factors</th>
<th>DK ∈ KB</th>
<th>DK ∈ KB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hidden factors</td>
<td>S-R</td>
<td>U-R</td>
</tr>
<tr>
<td>Direct relatedness</td>
<td>S-UR</td>
<td>U-UR</td>
</tr>
<tr>
<td>Unknown</td>
<td>S-UR</td>
<td>U-UR</td>
</tr>
<tr>
<td>Different facts in data</td>
<td>S-UR</td>
<td>U-UR</td>
</tr>
</tbody>
</table>

Table 6.2: Possible factors for different interpretations from a user. The acronyms are: S=‘Surprising,’ U=‘Unsurprising,’ R=‘Reasonable,’ and UR=‘Unreasonable.’

In summary, the possible factors underlying an explanation’s reasonableness are: (1) direct relatedness, (2) hidden factors, (3) unknown factors, and (4) different facts in data. The relationship between these factors and the four possible cases of a user’s response are listed in Table 6.2. The column DK ∈ KB means that a user’s knowledge of a rule is properly captured in KB so that the rule would be regarded as surprising; because the rule is unexpected to the KB and the KB properly represents the user’s knowledge, therefore the rule would be surprising to the user. Conversely, the column DK ∈ KB means that a user knows more about the rule than KB so that the rule would be regarded as expected. In terms of unexpectedness, if a user could not explicate a rule, and the generated explanation does not make sense, U-UR case, the rule should be unknown to the user.

6.3.2 Ranking Function

It is possible for the proposed method to generate multiple explanations. If there are too many explanations, they would overwhelm a user. Therefore, there is a need for a ranking function to highlight explanations of interest. We propose a ranking function based on the concept of ‘strong evidence’.

The reason why we decide not to use MDL (minimum description length) and MML (minimum message length) criteria, which are broadly used in the structure learning of BNs, is because the size of explanations is so small that Occam’s Razor is not a major concern.
6.3. EXPLANATION-GENERATING ALGORITHM

Strong Evidence

‘Strong evidence’, as its name suggests, is an evidence that strongly associated, e.g. high conditional probability, with the outcome. Based on the assumption that a user would prefer most probable evidence in a reasoning chain, we recommend it could be a measure that represents the interest of users.

Example 3 (Wet Grass.). As a pseudo example in the home automation domain, suppose there is an association rule:

\[ \text{Thunder} = T \rightarrow \text{WetGrass} = T(85\%) \]

stating that the grass is wet when thundering. Suppose there are two explanations found:

1. \( \text{Thunder} = T \rightarrow \text{Rain} = T(83\%) \rightarrow \text{WetGrass} = T(84.7\%) \)

2. \( \text{Thunder} = T \rightarrow \text{IndoorLight} = \text{ON}(57\%) \rightarrow \text{WetGrass} = T(85.4\%) \)

The occurrence of raining when thundering (83\%) is much higher than the occurrence of \( \text{IndoorLight} = \text{ON} \) (57\%), therefore \text{Rain} is a strong evidence. A user would naturally prefer the first explanation because raining is a more probable event when thundering.

The score of ‘strong evidence’ can be calculated directly from the CPT of BNs which represent explanations. The definition of the ranking function only focuses on the intermediate variable(s) \( X_i \) for the reason that the purpose of explanations is to find additional intermediate variable(s).

Definition 9 (Ranking Function). Given an explanation, \( e \), we define a ranking function, \( f_{\text{ranking}} \), that outputs the score of strong evidence.

\[
f_{\text{ranking}}(e) = \frac{1}{|X|} \sum_{i \in X} \frac{P_{\max}(X_i)}{P_{\text{even}}(X_i)}, \tag{6.9}
\]

where

\[
P_{\max}(X_i) = \max(P(X_i = x_j | A = a)),
\]

and \( P_{\text{even}}(X_i) \) is the probability of \( X_i = x_j \) under even distribution:

\[
P_{\text{even}}(X_i) = \frac{1}{|X_i|}.
\]
Example 4 (Explanations for Wet Grass.). Continue the previous example, the score of the two explanations are:

\[ e_1 : Thunder = T \rightarrow Rain = T \text{(83%)} \rightarrow WetGrass = T \text{(84.7%)}, \]

\[ f_{\text{ranking}}(e_1) = \frac{0.83}{0.5} = 1.66; \]

\[ e_2 : Thunder = T \rightarrow IndoorLight = ON \text{(57%)} \rightarrow WetGrass = T \text{(85.4%)}, \]

\[ f_{\text{ranking}}(e_2) = \frac{0.57}{0.5} = 1.14. \]

Therefore, Rain is a stronger evidence with higher score.

6.4 Related Work

Because the proposed ‘explanations’ are modeled by BNs with the concept of ‘intermediate variables’, they resemble the ‘causal diagrams’ which are broadly studied in the field of epidemiology. Further, the nature of finding ‘explanations’ is actually a process of structure learning of BNs. We briefly compare the proposed method with these two topics.

6.4.1 Causal Diagrams in Epidemiology

The causal diagrams (causal graphs/graphical models) are one major type of causal models for health-science research (Greenland and Brumback, 2002). A causal diagram is represented in a directed acyclic graph (DAG) which has the same definitions of the DAGs of Bayesian networks. The difference between causal diagrams and general BNs is in the interpretation of arrows – a graph is causal if every arrow represents the presence of an effect of the parent (causal) variable on the child (affected) variable; when BNs are built based on causal graphs, they are called as ‘causal Bayesian networks’ (Pearl, 2000, chap. 1).

Causal diagrams can serve as a visual yet logically rigorous aid for summarizing assumptions about a problem and for identifying variables that must be measured and controlled to obtain unconfounded effect estimates given those assumptions; thus, use of such graphs can aid in planning of data collection and analysis, in communication of results, and in avoiding subtle pitfalls of confounder selection (Greenland et al., 1999). Causal diagrams can also be used to estimate ‘direct effects’ (Petersen et al., 2006) or to control ‘confounders’ or ‘se-
6.4. RELATED WORK

lection bias’ (Hernán et al., 2004) in epidemiology. A direct effect (could be disease or outcome) of a variable (could be an exposure/intervention/treatment) is denoted by an arc/edge in a DAG. The other variables (other than cause and effect nodes) of a causal graph could be confounders (common causes), intermediate variables (intermediates), or selection biases (common effects).

Figure 6.5 shows an example of a causal diagram; Petersen et al. (2006) used different types of lines (e.g. solid and dashed lines) to represent indirect or direct effects; but it is more common to use only solid lines in DAGs in the literature. In the diagram, the confounder (C) affects both the intermediate (Z) and the outcome (Y); meanwhile, (C) is itself a causal intermediate between the exposure (A) and intermediate (Z). Node (Z) is an intermediate because it is on the causal path from exposure (A) to outcome (Y). Simply speaking, we can say that X and Y are confounded when there is a third variable Z (common cause) that influences both X and Y; such a variable is then called a ‘confounder’ of X and Y. In epidemiology, confounders are factors (exposures, interventions, treatments, etc.) that explain or produce confounding (Rothman and Greenland, 1998, p. 62). Any factor that represents a step in the causal chain between exposure and disease should not be treated as a confounding factor, but instead requires special treatment as an intermediate factor (Robins, 1989).

![Causal Diagram](image)

**Figure 6.5:** Example of causal diagram of confounding by causal intermediate (Petersen et al., 2006).

The proposed ‘explanations’ differ from causal diagrams in three aspects – namely causality, structure and modeling. First, as justified in Subsection 6.3.1, ‘explanations’ represent only associations; but causal diagrams attempt to model causality. Greenland et al. (1999) cautioned that ‘... we must emphasize that no approach solves the central epistemologic problem of inferring
causation from non-experimental (observational) data. As realized by Hume centuries ago and reinforced by many authors since, all causal inference is based on assumptions that cannot be derived from observations alone.’ Therefore the general settings of KDD resist the treatment of causality because the data is often observational.

In terms of the structures of ‘explanations’ and causal diagrams, the ‘explanations’ assume that the intermediate variables $X_i$ make $A$ and $B$ conditional independent; thus, there would be no edges between $A$ and $B$. On the contrary, causal diagrams attempt to determine direct effects between exposures ($E$) and outcomes ($Y$); an arc/edge between $E$ and $Y$ is of interest.

The ‘finding’ of explanations is simply defined based on the similarity of inferred conditional probability. However, the identification of causations in epidemiology is a much more complex and rigorous task: the experiment method has to be correctly decided, such as clinical trials, field trials, cohort studies, case-control studies (Rothman and Greenland, 1998); in the experiment and its analysis, a large portion of efforts is to avoid or adjust (control) for confounding (Robins, 1989; Greenland et al., 1999; Cole and Hernán, 2002; Hernán et al., 2004; Petersen et al., 2006); the statistical tools for measuring dependencies are more sophisticated than the one used in the ‘explanations’, e.g. multi-variable regression and associational/causal risk ratios.

6.4.2 Structure Learning of Bayesian Networks

The structure learning of Bayesian networks aims to rebuild a best (to a given metric) model from data. Here we briefly discuss two kinds of learners: (a) conditional independence learners and (b) Bayesian metric learners.

The conditional independence learners, e.g. the CI algorithm (Verma and Pearl, 1990a) and the PC algorithm of TETRAD II (Spirtes et al., 2000, chap. 5), apply statistical tests to detect conditional independence (as discussed in 6.2.2) for determining edges between variables. It is also possible to combine independence tests with mutual information metrics as a scoring function for the learning (de Campos, 2006). The conditional independence learners utilise the $d$-separation (dependence-separation) criterion to decide whether $X$ and $Y$ are directly connected: if there was any set of variables that makes $X$ and $Y$ being $d$-separated, then $X$ and $Y$ are not directly connected. Please refer to the Bayesian nets literatures, e.g. (Pearl, 2000; Korb and Nicholson, 2003),
for the formal definition of \( d \)-separation.

The second approach, Bayesian metric learners, tackles the learning problem by searching the model space \( \{h_i\} \), with some metric like \( P(\cdot|e) \) (\( e \) denotes evidence) aiming to select an \( h_i \) that maximises the function. So, there are two computationally difficult tasks these learners need to perform: (a) scoring individual hypotheses and (b) search the space of models which is exponential (Korb and Nicholson, 2003, chap. 8). This approach usually employees some heuristics, e.g. greedy, MDL (minimum description length), MML (minimum message length), for the searching in the exponential search space. The K2 algorithm of Cooper and Herskovits (1992) aims to find \( h_i \) which maximises \( P(h_i|e) \); the calculation of \( P(h_i|e) \) is derived to estimate the joint probability \( P(h_i, e) \) by the Baye’s theorem; with some simplifying assumptions, computing \( P(h_i, e) \) has become a straightforward counting problem which is equal to \( P(h_i) \) times a simple function of the number of assignments to parent and child variables and the number of matching cases in the sample. We point to - but omit the details of - various heuristics and metrics in previous research, such as hill-climbing (Heckerman et al., 1995), MDL (Lam and Bacchus, 1994; Suzuki, 1999), CaMML (Korb and Nicholson, 2003, chap. 8), BNC-\( n \)-P by conditional log likelihood and BNC-MDL (Grossman and Domingos, 2004).

The proposed definition and algorithm for explanation generation is fundamentally similar with previous research of structure learning of BNs; but is different in details in two aspects. First, the structure learning attempts to model the whole set of variables, but ‘explanations’ only focus on the variables of the rule and possible intermediates; thus, the search space is different: ‘explanations’ have restrictive and comparatively smaller search space of potential hypotheses/structures. Secondly, the flavor of metrics is different; structure learning estimates the likelihood of a hypothesis given evidence/data or the accuracy of prediction, but ‘explanations’ prefer the similarity of inference.

### 6.5 Case Studies

The first case study demonstrates the proposed method on the WWW User Survey dataset; two association rules are investigated in detail. In the second case study, we integrate the method of explanation generation into unexpected pattern mining (as Figure 6.6 shows); the overall process is compared with the method of (Jaroszewicz and Simovici, 2004) again.
6.5.1 Case Study I: WWW User Survey

In this Subsection we describe an empirical test of our method of finding explanations on Internet Use survey data. We apply the Apriori algorithm from the Orange software package\(^2\) for mining association rules. Two mined rules are shown in Table 6.3.

We select rule (1): WillingToPay=PreferFreeSources \(\rightarrow\) Gender=Male and find its explanation to see possible factors for this rule. The probability distribution of the rule is listed below.

<table>
<thead>
<tr>
<th>WillingToPay</th>
<th>P(Male)</th>
<th>P(Female)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PreferFreeSources</td>
<td>0.637</td>
<td>0.363</td>
</tr>
</tbody>
</table>

\(^2\)http://www.ailab.si/orange/
Table 6.3: Two mined association rules.

<table>
<thead>
<tr>
<th>rule</th>
<th>support</th>
<th>confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) WillingToPay = PreferFreeSources → Gender = Male</td>
<td>0.101</td>
<td>0.637</td>
</tr>
<tr>
<td>(2) Gender = Male, Occupation = Computer → WillingToPay = PreferFreeSources</td>
<td>0.102</td>
<td>0.539</td>
</tr>
</tbody>
</table>

One explanation for the rule is plotted in Figure 6.7 that ever ordering products or services using the web is a factor. The conditional probabilities of the explanation are listed below.

<table>
<thead>
<tr>
<th>WillingToPay</th>
<th>P(WebOrdering=Yes)</th>
<th>P(WebOrdering=No)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PreferFreeSources</td>
<td>0.78</td>
<td>0.216</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>WebOrdering</th>
<th>P(Male)</th>
<th>P(Female)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>0.662</td>
<td>0.338</td>
</tr>
<tr>
<td>NO</td>
<td>0.513</td>
<td>0.487</td>
</tr>
</tbody>
</table>

At first pass, our interpretation of the explanation is: if users are unwilling to pay for content because of other free content sources, they are also likely to have the experience of ordering products or services using the web; and when the users have experience in web ordering, they are likely to be male. Altogether, the factor WebOrdering could be an intermediate factor when we reason the confidence of the rule.

\[
\text{PreferFreeSources} \rightarrow \text{WebOrdering} \rightarrow \text{Male}
\]

Figure 6.7: One explanation for the rule WillingPay = PreferFreeSources \(\rightarrow\) Gender = Male. Score of explanation \(f_{\text{ranking}} = 1.56\).

The second result (Figure 6.8) is an explanation for rule (2) that when male users’ occupation is computer-related, they are more likely to find free sources of information than pay for it. The method finds that the user’s occupation and his experience in creating web pages can be the intermediate variables for explaining the rule.

Comparison

We implement the algorithm proposed by Yao et al. (2003) and apply it to the mined rule (1): WillingToPay = PreferFreeSources \(\rightarrow\) Gender =
Male $\rightarrow$ Job $\rightarrow$ PreferFreesources

ComputerOccupation $\rightarrow$ WebPageCreation

Figure 6.8: One explanation for the rule Gender=Male, Occupation=Computer $\rightarrow$ WillingPay=PreferFreesources. Score of explanation $f_{\text{ranking}} = 7.58$.

Male. The top 6 explanations are listed in Table 6.4. The found conditions are all about variables Job and Country where these conditions can increase the support of conditional association, $s(\phi\psi|\chi) > s(\phi\psi)$. However, we can see that these conditions only cover 19 instances of rule (1) while there are still 2658 instances not covered by these explanations. Compared to our approach, the total 2677 instances of rule (4) are all covered by the explanation WillingPay $\rightarrow$ WebOrdering $\rightarrow$ Gender in terms of probability distribution.

| Condition $\chi$                                | $s(\phi\psi|\chi)$ | Covered instances |
|------------------------------------------------|---------------------|-------------------|
| Job = Other $\wedge$ Country = SouthKorea       | 1                   | 1                 |
| Job = Other $\wedge$ Country = DominicanRepublic| 1                   | 1                 |
| Job = Other $\wedge$ Country = Egypt            | 1                   | 1                 |
| Job = Other $\wedge$ Country = Greece           | 1                   | 1                 |
| Job = Other $\wedge$ Country = Sweden           | 0.667               | 12                |
| Job = Other $\wedge$ Country = Portugal         | 0.667               | 3                 |

6.5.2 Case Study II: Census Income

This case study proceeds as described. At first, we construct the initial Bayesian network, $BN_{t=0}$, using our common-sense knowledge. Of course, this network would be more informative if the user was a domain expert. Second, a collection of dependence rules is mined. Then we mine unexpected dependence rules, $R_U$, based on $BN_{t=0}$. At fourth step, we find possible explanations, $E_{r_U}$, for each $r \in R_U$. At the final step of this iteration, we modify

---

3 The condition $\chi$ is discovered by the C4.5 decision tree algorithm.
the structure of the Bayesian network as $BN_{t=1}$ based on $R_U$ and $E_{ru}$ and go to third step for next iteration. The method of Jaroszewicz and Simovici (2004) is implemented and applied as well for comparison. The initial Bayesian network, $BN_{t=0}$, is assigned with six edges: (edges 1 to 4) Education and Education-Number to Work and Occupation and (edges 5 and 6) Work and Occupation to Income. Figure 6.9 shows the initial Bayesian network, $BN_{t=0}$.

Figure 6.9: The initial Bayesian network constructed by our common-sense knowledge.

**First Iteration**

**Mining Unexpected Rules**

We input the data and $BN_{t=0}$ to algorithm 2. The top 9 unexpected rules are shown in Table 6.5.

**Finding Explanations**

Examining the first unexpected rule: \{Education = Prof-school, CapGain $\geq$ 15022.0\}, we apply the proposed method for finding possible explanations. It returns three explanations under $\varepsilon_F = 0.001$ which are shown in Figure 6.10. We can see that Education-Number, Marital, Relationship and Occupation are four possible factors that make this rule reasonable. These factors somewhat match our common sense; for example, people who are in specific occupations invest well and different educations lead to different occupations.

Examining the second rule: \{Marital = Married-spouse-absent, occupation = Farming-fishing\}, we are surprised that farmers and fishers are connected to the absence of spouses. We set $A$=Occupation and $B$=Marital and find possible explanations for this rule (Figure 6.11). Our common sense tells us that Work (Work Class) is related to Occupation and Relationship is related
Table 6.5: Mined unexpected rules with $\epsilon_U = 4$

<table>
<thead>
<tr>
<th>Rule</th>
<th>interest</th>
<th>$U$</th>
</tr>
</thead>
<tbody>
<tr>
<td>{Education=Prof-school, CapGain $\geq 15022.0$}</td>
<td>10.42</td>
<td>10.42</td>
</tr>
<tr>
<td>{Marital=Married-spouse-absent, Occupation=Farming-fishing}</td>
<td>1.98</td>
<td>7.49</td>
</tr>
<tr>
<td>{Relationship=Other-relative, Native=Guatemala}</td>
<td>7.0</td>
<td>7.0</td>
</tr>
<tr>
<td>{Education=5th-6th, Marital=Married-spouse-absent}</td>
<td>5.66</td>
<td>5.66</td>
</tr>
<tr>
<td>{Work=Self-emp-not-inc, Occupation=Machine-op-inspect}</td>
<td>0.21</td>
<td>5.41</td>
</tr>
<tr>
<td>{Work=Self-emp-inc, CapGain $\geq 15022.00$}</td>
<td>5.2</td>
<td>5.2</td>
</tr>
<tr>
<td>{Education=5th-6th, Race=Other}</td>
<td>4.98</td>
<td>4.98</td>
</tr>
<tr>
<td>{Marital=Married-spouse-absent, Race=Other}</td>
<td>4.94</td>
<td>4.94</td>
</tr>
<tr>
<td>{Education=Doctorate, CapLoss=$1881.5 \sim 1975.5$}</td>
<td>4.8</td>
<td>4.8</td>
</tr>
<tr>
<td>{EducationNum $\geq 13.00$, Native=India}</td>
<td>5.69</td>
<td>4.8</td>
</tr>
<tr>
<td>{Education=5th-6th, Relationship=Other-relative}</td>
<td>4.59</td>
<td>4.59</td>
</tr>
</tbody>
</table>

$$ProSchool \rightarrow EducationNum \rightarrow CapGain \geq 15022.0$$

$$Occupation \quad (f_{\text{ranking}} = 13.885)$$

$$ProfSchool \rightarrow EducationNum \rightarrow CapGain \geq 15022.0$$

$$Marital \quad (f_{\text{ranking}} = 10.503)$$

$$ProfSchool \rightarrow Marital \rightarrow CapGain \geq 15022.0$$

$$Relationship \quad (f_{\text{ranking}} = 4.505)$$

Figure 6.10: Possible explanations for the rule: \{Education = Prof-school, CapGain $\geq 15022.0$\}

In comparison, the unexpected frequent itemsets (Jaroszewicz and Simovici, 2004) are mined and the result is shown in Table 6.6. Note that the interestingness measure in Table 6.6 is different from the ‘interest’ of definition 4; it is defined by (Jaroszewicz and Simovici, 2004) (shown in Equation 2.9). Comparing Table 6.5 and Table 6.6, the method of (Jaroszewicz and Simovici, 2004)
6.5. CASE STUDIES

Figure 6.11: Possible explanations for the rule: \{Marital = Married-spouse-absent, occupation = Farming-fishing\}. Scores of explanations are 5.082, 2.98, 1.768 respectively.

found two associations considerably fit our common-sense: Native → Race and CapitalGain → CapitalLoss. Although the unexpected dependence rules do not reveal these two associations, they give other interesting and specific patterns. For example, the first rule indicates that people who are graduated from professional school are very likely to earn money in capital investment.

Table 6.6: Unexpected frequent itemsets Jaroszewicz and Simovici (2004) with $\epsilon = 0.01$

<table>
<thead>
<tr>
<th>Rule</th>
<th>Interestingness</th>
</tr>
</thead>
<tbody>
<tr>
<td>{ Race, Native }</td>
<td>0.02</td>
</tr>
<tr>
<td>{ Race, Capital Loss }</td>
<td>0.012</td>
</tr>
<tr>
<td>{ Capital Gain, Capital Loss }</td>
<td>0.012</td>
</tr>
</tbody>
</table>

Updating the Bayesian Network

For demonstration purpose, we assume that we have no doubt about unexpected rules 3 to 11. We therefore update the Bayesian network according to the rules 3 to 11 and the explanations of the first two rules. For example, we add links: Occupation → CapitalGain, Income → Marital and Native → Relationship. The updated Bayesian network, $BN_{t=1}$, is plotted in Figure 6.12. The comparative Bayesian network updated by the method in Jaroszewicz and Simovici (2004) is also shown in Figure 6.13.

Second Iteration

Mining Unexpected Rules

Since the Bayesian network is updated, we can perform another iteration for finding new unexpected rules and their explanations. Again, we input data
and $BN_{t=1}$ to algorithm 5 with $\epsilon_U = 3$. The top 6 unexpected rules are listed in Table 6.7. The first rule is the same as the first rule in previous iteration; this means that current Bayesian network still cannot fully represent the joint probability for this rule. This case also happens to the second rule that $Occupation$ and $Marital$ was connected via $Income$. Because $BN_{t=1}$ tries to model the first two rules via intermediate variables but it does not succeed, we suspect $\{Education, CapGain\}$ and $\{Occupation, Marital\}$ are truly connected.

**Finding Explanations**

Because the interest of the third rule is 0.02, it means the joint probability of $CapGain = 7073.5 \sim 15022$ and $Income \leq 50K$ is much less than the product of their marginal probability: they have a negative dependence. We cannot find any explanation for this rule with $\epsilon_E = 0.01$. This indicates that the factors of this rule are not observed in data.

The explanations for the fourth rule, $\{EducationNum \geq 13, CapGain \geq 15022\}$, are listed in Figure 6.14. We can see that $Education$, $Occupation$ and $Gender$ are possible factors for this rule.
Table 6.7: Mined unexpected rules in second iteration with $\epsilon_U = 3$

<table>
<thead>
<tr>
<th>Rule</th>
<th>interest</th>
<th>$U$</th>
</tr>
</thead>
<tbody>
<tr>
<td>{Education=Prof-school, CapGain$\geq$15022.0}</td>
<td>10.42</td>
<td>6.08</td>
</tr>
<tr>
<td>{Marital=Widowed, Occupation = Priv-house-serv}</td>
<td>5.61</td>
<td>4.71</td>
</tr>
<tr>
<td>{CapGain=7073.5$\sim$15022, Income$\leq$50K}</td>
<td>0.02</td>
<td>3.71</td>
</tr>
<tr>
<td>{EducationNum$\geq$13, CapGain$\geq$15022}</td>
<td>4.83</td>
<td>3.62</td>
</tr>
<tr>
<td>{Marital=Separated, Relationship=Unmarried}</td>
<td>4.13</td>
<td>3.58</td>
</tr>
<tr>
<td>{Occupation=Priv-house-serv, Relationship=Other-relative}</td>
<td>4.5</td>
<td>3.47</td>
</tr>
<tr>
<td>{Work=Self-emp-inc, WorkHours$\geq$59.50}</td>
<td>3.29</td>
<td>3.29</td>
</tr>
<tr>
<td>{Age$\geq$50.50, Marital=Widowed}</td>
<td>3.97</td>
<td>3.22</td>
</tr>
</tbody>
</table>

$\text{EducationNum} \geq 13 \rightarrow \text{Education} \rightarrow \text{CapGain} \geq 15022$

Figure 6.14: Possible explanations for the rule: \{EducationNum$\geq$13, CapGain$\geq$15022\}. Scores of explanations are 10.621, 8.203, 6.02 respectively.

Algorithm 5 cannot find any explanation for the fifth rule: \{Marital=Separated, Relationship=Unmarried\}. The explanations of the sixth rule, \{Occupation=Priv-house-serv, Relationship=Other-relative\}, are listed in Figure 6.15. It implies that there are multiple factors and effects for this rule and Native is the most common factor.

Comparison

Again, the unexpected frequent itemsets Jaroszewicz and Simovici (2004) are mined according to Figure 6.13 and the result is shown in Table 6.8. These unexpected frequent itemsets are very similar to those in Table 6.6. They are all about Native, Race, CapGain, and CapLoss but there are no new variables discovered.

Updating the Bayesian Network

According to the explanations in Figures 6.14 and 6.15, we decide to add these links: $\text{Education} \rightarrow \text{CapGain}$ and $\text{Education} \rightarrow \text{EducationNum}$, Native $\rightarrow$
CHAPTER 6. EXPLANATION GENERATION

\[
\text{PrivHouseServ} \rightarrow \text{Native} \rightarrow \text{OtherRelative} \quad (f_{\text{ranking}} = 25.369)
\]

\[
\text{PrivHouseServ} \rightarrow \text{Marital} \rightarrow \text{OtherRelative}
\]

\[
\text{Native} \quad (f_{\text{ranking}} = 14.253)
\]

\[
\text{PrivHouseServ} \rightarrow \text{Age} \rightarrow \text{OtherRelative}
\]

\[
\text{Native} \quad (f_{\text{ranking}} = 13.552)
\]

\[
\text{PrivHouseServ} \rightarrow \text{Work} \rightarrow \text{Gender} \rightarrow \text{OtherRelative} \quad (f_{\text{ranking}} = 5.158)
\]

\[
\text{PrivHouseServ} \rightarrow \text{Education} \rightarrow \text{OtherRelative}
\]

\[
\text{Income} \quad (f_{\text{ranking}} = 3.677)
\]

Figure 6.15: Possible explanations for the rule: \{Occupation=Priv-house-serv, Relationship= Other-relative\}.

Table 6.8: Unexpected frequent itemsets Jaroszewicz and Simovici (2004) with \(\epsilon = 0.007\)

<table>
<thead>
<tr>
<th>Rule</th>
<th>Interestingness</th>
</tr>
</thead>
<tbody>
<tr>
<td>{ Capital Gain, Capital Loss, Native }</td>
<td>0.022</td>
</tr>
<tr>
<td>{ Race, Capital Loss, Native }</td>
<td>0.019</td>
</tr>
<tr>
<td>{ Race, Capital Gain}</td>
<td>0.007</td>
</tr>
</tbody>
</table>

Occupation and

Native \rightarrow Relationship. The other new links are added according to rules 1,2,3,5,7 and 8. The updated Bayesian network, \(BN_{t=2}\), is plotted in Figure 6.16. On the other hand, we add links – Race \rightarrow CapGain, Native \rightarrow CapGain and Native \rightarrow CapLoss – for the comparative network, which are plotted in Figure 6.17.

Summary

We summarize the overall results of this section. Figure 6.9 shows the manually constructed Bayesian network as the initial network for mining unexpected rules. Tables 6.5 and 6.7 are the mined unexpected dependence rules in the two iterations. Figures 6.10, 6.11, 6.14 and 6.15 are the possible explanations
6.6. Chapter Summary

This chapter presents a probabilistic approach for generating explanations from data. The basic modeling of an explanation is a reasoning chain, then it is extended to be modelled by BNs. The concept is to find additional factors (from the attributes in data) that can model similar probability distribution to the original rule. The complexity of explanation generation is subject to the size of additional factors; in the thesis, it is tractable because we limit the number of additional factors to two. Due to the nature of data mining, the proposed method should be incapable of confirming cause and effect; in fact, it could only recover associations (Subsection 6.3.1). An analysis of the implications of a user’s judgement on a rule’s surprisingness and an explanation’s reasonableness is presented in Subsection 6.3.1. We also propose a ranking function for sorting the explanations of a rule in the flavor of ‘strong evidence’ (Subsection 6.3.2). The proposed ‘explanations’ are compared with the related work of causal diagrams and structure learning of BNs in Section 6.4. Finally two case
studies are presented in Section 6.5 to demonstrate the proposed method and its integration into unexpectedness.
Chapter 7

Case Studies

In this chapter, two case studies are used to evaluate the proposed methods for mining surprising patterns described in chapter 6, using real-world data and domain experts. The first case study is in the medical domain: specifically, clinical treatment of chronic renal disease patients (ANZDATA). We apply the proposed design of unexpectedness to mine unexpected patterns, and find explanations for these patterns; then present the result to a nephrologist to find the existence of surprising rules. The second case study is in the educational domain: specifically, misconceptions of decimal notation by elementary students (DCT data). Since the Decimal Comparison Test (DCT) was designed by the domain experts who participate this case study, the experts were able to provide a ‘complete’ Bayesian network representing their DK prior to the experiment, allowing evaluation of the performance of the proposed methods when KB closely represent the experts’ DK.

7.1 Case Study I: Nephrology

A working clinical nephrologist provided the domain expertise. The aim of the case study was to find patterns that were surprising to the expert, with the longer term ambition of potentially triggering new questions for research in renal disease. From experience, continuing education, and regular reference to the official annual reports of ANZDATA, the expert already had a deep understanding of the prevalence, trends, risk factors, etc. in the data. The case study is carried out by the procedure of Figure 6.6.

Data Preprocessing
We preprocess the dataset in three aspects: attribute selection, entry merging and discretisation\(^1\). In terms of attribute selection, some meaningless attributes for resulting rules (such as the sequence number of each entry in the database) are removed. The expert indicates that attributes of cardiovascular diseases and mortality are of interest, therefore, all disease and mortality related attributes are kept. There are many paired attributes with the same meaning but different encoding; from each pair we choose one of them for the new dataset. In terms of entry merging, we merge patient’s multiple entries into one row because we do not tackle temporal relations in the experiment. The entry merging procedure, however, is a compromise in the data mining setting because much of the expert’s knowledge consists of temporal relationships; this situation was learned during discussing the results with the expert (see Subsection 7.3.4). Finally, the discretisation process equally divides numerical attributes into 4 bins by their values. The preprocessed new dataset has 39 attributes; several of the discretised attributes are shown in Table 7.1. The discretisation procedure, which is based on equal division of the range of values (rather than, for instance, the number of entries), overlooks the importance of domain knowledge, even in what seems so simple a preprocessing step, because data preprocessing step is out of the scope of our overall design of unexpected rules mining. Notably, the discretisation issue turns out to be a gap in surprisingness (see Subsection 7.3.4). In this experiment, nevertheless, we use naive solutions for data preprocessing because we have not formally researched it; in fact, our design focuses on the steps of post-processing and pattern interpretation.

**Knowledge Acquisition**

As there are 39 variables in the preprocessed dataset, it would be too disorderly to visualise the complete graph for the expert to assign associations. Instead, a table was devised, in which each variable is located in the first column and the second column is empty so that the user can assign relates-to variables.

\(^{1}\)The term ‘preprocess’ we use here actually consists of data selection, preprocessing and transformation under the framework of Fayyad et al. (1996).
Table 7.1: Several of the discretised attributes in the preprocessed ANZDATA.

By working with the table, the expert input a rich AG that provides 375 links among the 39 variables (Table 7.2 shows a fragment of it); it is 49.3% connected in comparison to a complete graph. Because the acquired AG is cyclic, we then process it by the method proposed in Subsection 5.3.3 into DAG for the building of BNs.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Relates to</th>
</tr>
</thead>
<tbody>
<tr>
<td>agestart</td>
<td>agequart, dryweight, height, weightb, txwlcat, diabetes, diseaseb, presentt, flowrate, modality, referralb, biopsy, cadb</td>
</tr>
<tr>
<td>dryweight</td>
<td>agequart, bmib, height, weightb, hourst, flowrate</td>
</tr>
<tr>
<td>height</td>
<td>agequart, bmib, weightb</td>
</tr>
<tr>
<td>raceb</td>
<td>agequart, agestart, bmib, biopsy, modality, referralb, cig, cadb, cvdb, diabetes, diseaseb, pvd, lungb, hypertension, vascdissb, agedeath, alldeath, cardeath, c ausdethb, circdeath, cvdeath, hfddeath, mideath, weightb</td>
</tr>
<tr>
<td>sex</td>
<td>bmib, height, weightb, cig, cadb, cvdb, diabetes, hypertension, lungb, pvd, vascdissb, agedeath, alldeath, cardeath, causdethb, circdeath, cvdeath, hfddeath, mideath, flowrate, diseaseb</td>
</tr>
</tbody>
</table>

Table 7.2: A fragment of the table representing the association graph of ANZDATA assigned by the medical expert.

Dependent Association Rules
Orange data mining software (Demsar and Zupan, 2004) was then used to mine association rules with minimal \( \text{support} = 0.6 \) and \( \text{confidence} = 0.6 \); there are 1279 rules mined (see Appendix G for example). We acknowledge that the \( \text{support} = 0.6 \) is unusually high comparing to the general practices of association rules mining where \( \text{support} \) is usually set to 0.1 even further
to 0.01. Actually the choice of \( \text{support} = 0.6 \) is a compromise due to the memory limitation under the size of ANZDATA and the requirement of Orange software. In another trial of the project, but not reported in the thesis, we implemented association rule mining code by ourself to enable us to mine ARs at \( \text{support} = 0.01 \); in this case 75,377 rules were mined.

In the second step we verify the independence of every association rules via chi-squared test as Subsection B.1 introduced. There are 361 rules which passed the \( \chi^2 \) test being dependent under significance level \( \alpha = 0.05 \).

**Unexpected Rules**

The mined dependent association rules are then measured for their unexpectedness values by the algorithm discussed in subsection 5.3.1. The top 6 unexpected rules are listed in Table 7.3.

<table>
<thead>
<tr>
<th>No.</th>
<th>Unexpected Rules</th>
<th>support</th>
<th>confidence</th>
<th>( U )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>cvdb=No ( \rightarrow ) dryweight=53.2-101.5</td>
<td>0.64</td>
<td>0.70</td>
<td>0.09</td>
</tr>
<tr>
<td>2</td>
<td>mideath=AliveorNon-MIdeath ( \rightarrow ) dryweight=53.2-101.5</td>
<td>0.62</td>
<td>0.70</td>
<td>0.09</td>
</tr>
<tr>
<td>3</td>
<td>lungb=No ( \rightarrow ) dryweight=53.2 101.5</td>
<td>0.63</td>
<td>0.71</td>
<td>0.09</td>
</tr>
<tr>
<td>4</td>
<td>lungb=No, hfdeath = AliveorNon-HFdeath ( \rightarrow ) txwlcat = NotonTransplantList</td>
<td>0.64</td>
<td>0.74</td>
<td>0.07</td>
</tr>
<tr>
<td>5</td>
<td>lungb=No ( \rightarrow ) txwlcat = NotonTransplantList</td>
<td>0.65</td>
<td>0.74</td>
<td>0.07</td>
</tr>
<tr>
<td>6</td>
<td>hfdeath=AliveorNon-HFdeath, mideath=AliveorNon-MIdeath ( \rightarrow ) txwlcat=NotonTransplantList</td>
<td>0.63</td>
<td>0.73</td>
<td>0.07</td>
</tr>
</tbody>
</table>

Table 7.3: Mined unexpected rules. \( U \) in the last column is unexpectedness.

Table 7.4 lists the expert’s feedback of unexpectedness about the mined rules.

As the result shows, the proposed design for unexpected rules mining successfully discovers surprising rules (rules 5 and 6). The local Bayesian networks of rule 5 and 6 are plotted in Figures 7.1 and 7.2. Rule 5 states that ‘patients without lung disease are also not on the transplant list’. The expert commented on rule 5 in the following:
7.1. CASE STUDY I: NEPHROLOGY

<table>
<thead>
<tr>
<th>Rule No.</th>
<th>Unexpectedness of rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,2,3,4</td>
<td>Meaningless</td>
</tr>
<tr>
<td>5</td>
<td>Surprising</td>
</tr>
<tr>
<td>6</td>
<td>Surprising</td>
</tr>
</tbody>
</table>

Table 7.4: User's feedback about unexpected rules.

Figure 7.1: Local BN for Rule 5, lungb=No → txwlcat = NotonTransplantList.

Figure 7.2: Local BN for Rule 6, hfdeath=AliveorNon-HFdeath, mideath=AliveorNon-MIdeath → txwlcat=NotonTransplantList.

“Rule 5 is surprising as we would expect these patients to be on the list if there are no other problems.”

Rule 6 states that patients who are alive or not die from heart failure and myocardial infarction are also not on the transplant list. The expert commented on rule 6 in the following:

“Rule 6 is also surprising as we would expect these patients to be on the list because they should be in good health conditions.”

Explanations
Here we apply the proposed method to explain the last two unexpected rules of Table 7.3. The local Bayesian network and an explanation of rule 5 are plotted in Figures 7.1 and 7.3. Rule 5 states that patients without lung disease are
also not on the transplant list. The expert commented on the explanation of rule 5 as:

“The generated explanation to Rule 5 relates late referral to no lung disease. A possible explanation is that the late referrals have more complications and higher risk of infection which may delay them being put on the transplant list, or may even be a permanent factor for why they are not on the list.”

The local BN and reasonable explanation of rule 6 are plotted in Figures 7.2 and 7.4. Rule 6 states that patients who are alive or not die from heart failure and myocardial infarction are also not on the transplant list. The expert commented on rule 6 in the following:

“The generated explanation suggests late referral being associated with these group of patients. It is possible that late referral and its associated risks may be a reason why patients are not on or are delayed being put on the transplant list. Interestingly, the explanation rediscover\textit{agequart} as an intermediate factor just as the same as the KB in Figure 7.2.”

\textbf{Interpretation Problems}

The expert thought rules 1 to 4 are somehow meaningless and hard to interpret
For rules 1 to 3, the **dryweight** variable is subject to age and sex, therefore, these rules are hard to interpret because **dryweight** itself along does not carry sufficient information. Furthermore, 53.2–101.5 kg is a normal range for most of the patients. Perhaps, overweight and underweight would be more meaningful. Therefore, the variable could be ruled out in data mining because we have the variable **BMI** (body mass index). Moreover, the rules do not have temporal information such as weight loss over time which is more meaningful to screen a patient’s progress.

“Rule 4 seems strange because of two reasons. First, when the patients do not have lung disease, they are likely to be fit and well and they tend to pass the screening process for the transplant waiting list; therefore, this part of the rule is unexpected. The second reason is that the heart failure death variable (**hfdeath**) makes the rule strange because death happens after being on the transplant list or not. In other words, doctors do not determine a patient being on the transplant list or not because of their cause of death.”
Due to the problems encountered, a decision was made to explore ways of improving the quality of mined rules before evaluating the idea of updating knowledgebase iteratively. Therefore, the case study of ANZDATA is ceased after the first round of the iterative design (Figure 6.6).

7.2 Case Study II: Education in Mathematics

The second case study has slightly contrasting circumstances with the first. The background of mining ANZDATA is that the expert is not completely familiar with the specific labeling and encoding of the dataset prior to the data mining, although they are completely familiar with the domain. On the other hand, the experiment of mining DCT (Decimal Comparison Test) data has the circumstance that the items of tests, the types of items and codes of results are all carefully designed by the domain experts who participated in this case study.

The diagnostic test, DCT, was created from a longitudinal study (Steinle, 2004) by extending and refining tests in the literature (Steinle and Stacey, 1998; Stacey and Steinle, 1999) to identify students with one of 12 misconceptions about decimal notation. In the study, over 3000 students, from a volunteer sample of 12 schools in Victoria, Australia, completed nearly 10000 tests over a four-year period. The experts also participated in the development of intelligent tutoring systems for decimal notation misconceptions (Nicholson et al., 2001; Boneh et al., 2004) where their DK was elicited into a Bayesian network. According to the experts, the dependencies of items, types and codes in the DCT data have one-to-one mapping to the structure of the BN. For example, the attribute $\text{CoarseCode}$ is a simplified representation of $\text{FineCode}$, so that the BN encodes this pairwise relation $\text{CoarseCode} \rightarrow \text{FineCode}$.

Knowledge Acquisition

Under this circumstance, the task of knowledge acquisition is straightforward by inputting the Bayesian network provided by the experts; Figure 7.5 shows the acquired KB that the attributes are $\text{Year}$, $\text{CoarseCode}$, $\text{FineCode}$, and $\text{Type}1$ to $\text{Type}6$. 
7.2. CASE STUDY II: EDUCATION IN MATHEMATICS

Dependence Rules and Unexpected Rules

Because the original goal of DCT is to identify the misconceptions of students in reasoning with decimal notation, it is reasonable to analyse the subset of students who make mistakes in the test; thus the records of 2331 out of 3531 students who had at least one error in the test are selected for data mining. Because we were researching unexpectedness for dependence rules at the time of the DCT case study, this experiment is based on dependence rules. Under the minimal interest = 0.1 and p-value of 0.05, there are 44 dependence rules mined.

In mining unexpected patterns, we gauge the unexpectedness of every dependence rules against the KB in Figure 7.5. The top five unexpected rules are listed in Table 7.5.

Table 7.5: Mined unexpected dependence rules with $\epsilon_U = 0.7$

<table>
<thead>
<tr>
<th>No.</th>
<th>Rule</th>
<th>interest</th>
<th>unexpectedness</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>{Type1 = M, Type2 = M}</td>
<td>1.84</td>
<td>2.91</td>
</tr>
<tr>
<td>2</td>
<td>{Type3 = M, Type4 = M}</td>
<td>2.26</td>
<td>1.31</td>
</tr>
<tr>
<td>3</td>
<td>{Year = y5, FineCode = LWH}</td>
<td>1.84</td>
<td>0.89</td>
</tr>
<tr>
<td>4</td>
<td>{Year = y10, CoarseCode = L}</td>
<td>0.36</td>
<td>0.79</td>
</tr>
<tr>
<td>5</td>
<td>{Year = y10, Type5 = L}</td>
<td>1.81</td>
<td>0.76</td>
</tr>
</tbody>
</table>

The interpretation of rule 1 \{Type1 = M, Type2 = M\} is that many stu-
dents who are at middle level of item Type 1 are often at middle level of item Type 2 as well. Similarly the interpretation of rule 2 \( \{ \text{Type3} = M, \text{Type4} = M \} \) is that many students who are at middle level of item Type 3 are often at middle level of item Type 4 as well. The interpretation of rule 3 \( \{ \text{Year} = y5, \text{FineCode} = \text{LWH} \} \) is that many students who are in fifth grade are often classified as LWH fine code. The interpretation of rule 4 \( \{ \text{Year} = y10, \text{CoarseCode} = \text{L} \} \) is that few students who are in tenth year are classified as L coarse code.

**Explanations**

The identified explanations of rules 1, 2 and 3 are described below.

![Figure 7.6: Possible explanation for the rule: \( \{ \text{Type1} = M, \text{Type2} = M \} \)](image)

The explanation in Figure 7.6 means that when the variables Type4 and Type6 are the intermediate variables, the probability \( P(\text{Type2} = M | \text{Type1} = M) \) can be precisely approximated by this Bayesian network. Without inspecting its CPT in detail, we can roughly infer that when students are at medium level of Type 1, they show some characteristic in Types 4 and 6 error that lead to the medium level of Type 2.

For rule 2, the explanation in Figure 7.7 indicates that when students are at medium level of Type 3, they show some characteristic in FineCode and Type2 that lead to the medium level of Type 4. Meanwhile, their performance on Type 2 influence their performance on Types 3 and 4.

![Figure 7.7: Possible explanation for the rule: \( \{ \text{Type3} = M, \text{Type4} = M \} \)](image)

For rule 3, we present two explanations to the experts in Figure 7.8. The two explanations suggest that having CoarseCode, Type5 and Type3
as intermediate variables, we can infer their probability of $P(F\text{ineCode} = LWH | Year = y5)$ precisely.

\[
\begin{align*}
\text{Year} & \rightarrow \text{CoarseCode} \rightarrow \text{FineCode} \\
& \quad \uparrow \text{Type5} \\
\text{Year} & \rightarrow \text{CoarseCode} \rightarrow \text{FineCode} \\
& \quad \uparrow \text{Type3}
\end{align*}
\]

Figure 7.8: Possible explanations for the rule: \{Year = y5, FineCode = LWH\}

The expert thought the rules 1 and 2 are worth for investigation because \textbf{M} value is inexplicable in their knowledge. They suggested to analyse further to the level of items/questions instead of item types. For rules 3 and 4, the experts already knew the relation between age (Year) and other variables but they did not model it in the BN for some reasons. Therefore rules 3 and 4 are expected.

In the experts’ opinion, the explanations of rules 1 and 2 (Figures 7.6,7.7) did not persuade them with reasonable factors. They acknowledged that there might be some correlations among these variables because their nature of definitions; we had further investigated the two explanations with the experts to understand their implication; the investigation is reported in the next paragraph. For the explanations of rule 3, the experts thought that the factor \textbf{CoarseCode} (in both explanations in Figure 7.8) is obvious because the \textbf{CoarseCode} is a rough classification of \textbf{FineCode}; thus, they suggested that \textbf{CoarseCode} should not be counted as a factor. However, we regard it is a good demonstration that ‘explanations’ could find ‘relevant’ factors.

**In-depth Investigation**

The presentation of the explanations in Figures 7.6, 7.7 did not provide sufficient information to the experts. Form the domain experts’ point of view, rules 1 and 2 (of Table 7.5) represent the unclassified (UN) group of students of whom the experts have little knowledge. Since this (UN) group is not understood by the experts, the distribution of other variables is also not known at the time of discussion (it turned out that there are some known dependence due to the definition of attributes); thus the experts had no prior knowledge
of the relations between variables of the explanations.

We plot the histogram of the explanation (Figure 7.6) for rule 1 in Figure 7.9; the histogram shows the distribution of Type4 and Type6 of the Type1=M \∧ Type2=M students. Among the 50 students, 8 students got Type6=1 \∧ Type4=4 whom the experts could explain:

‘... I can see 8 people who might get all the comparisons greater than 1.0 correct and all the comparisons less than 1.0 incorrect. This is probably because they think that numbers with a zero in the units column are less than zero, so they correctly compare and then swap the answer around as you would for negative numbers.’

According to the experts, the eight students was only incorrect in Q26 and Q28 because they incorrectly reason leading-zero numbers in the way of negative numbers. Table 7.6 highlights the difference between leading-zero items and the others.

Figure 7.9: The histogram of Type4 and Type6 in the first explanation.

The 9 students at another big bump Type6=3 \∧ Type4=0 in Figure 7.9 are the extreme where they got all correct in Type6 but all wrong in Type4. The ex-
7.2. CASE STUDY II: EDUCATION IN MATHEMATICS

<table>
<thead>
<tr>
<th>Item Type</th>
<th>Item</th>
<th>Questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type 4</td>
<td>Q21</td>
<td>4.4502/4.45</td>
</tr>
<tr>
<td>Type 4</td>
<td>Q22</td>
<td>17.353/17.35</td>
</tr>
<tr>
<td>Type 4</td>
<td>Q23</td>
<td>8.24563/8.245</td>
</tr>
<tr>
<td>Type 4</td>
<td>Q24</td>
<td>3.2618/3.26</td>
</tr>
<tr>
<td>Type 6</td>
<td>Q26</td>
<td>0.35/0.42</td>
</tr>
<tr>
<td>Type 6</td>
<td>Q27</td>
<td>2.186/2.954</td>
</tr>
<tr>
<td>Type 6</td>
<td>Q28</td>
<td>0.872/0.813</td>
</tr>
</tbody>
</table>

Experts suspected those students might make mistakes because they think shorter is bigger (for various reasons) or they treat the numbers as money so that they truncate the numbers to cents, e.g. convert 17.353/17.35 to $17.35/$17.35.

In terms of the second rule \{Type3 = M, Type4 = M\}, the experts indicated that these students must be careless. Based on the experts’ definition, these careless students are classified as unknown (U) categories in FineCode, e.g. UN, SU, LU and AU. Therefore, the presence of FineCode in the explanation (in Figure 7.7) could be viewed as a re-discovery of the innate definition of FineCode; this argument could be clearly perceived from the histogram of the explanation in Figure 7.10. The experts were expecting the FineCode could somehow provide additional information to the Type2 in the explanation; however, they could not perceive any interpretable cluster of the histogram in Figure 7.10. Nevertheless, the experts agreed on the finding of FineCode is reasonable because it represents the unknown class of students.

7.2.1 The Medium Level of Types 1 and 2

According to the experts’ suggestion, we then focused on the subset of the data of Type1=M ∨ Type2=M; there are 469 instances in the subset. In addition to item type nodes, the 30 items are also included; the variables are Year, CoarseCode, FineCode, Types 1 to 6 and items of these types. The initial BN is assigned partially that only the questions of types 1 and 2 are connected to Type1 and Type2 (Figure 7.11).

We left the items of types 3,4,5, e.g. Q3,Q4,Q5, not connected with their type nodes, and we expected them to appear in the unexpected rules; because we wanted to test whether the mining of unexpectedness could guide the recover of the structure of BN. The result confirms this expectation that
the highly unexpected rules are related to questions of types 3, 4, 5 and their corresponding types. Table 7.7 lists some of these rules.

Revisiting the goal of this section, we list the unexpected rules of Type1
7.2. CASE STUDY II: EDUCATION IN MATHEMATICS

Table 7.7: Mined unexpected dependence rules that confirms the missing link between questions and types.

<table>
<thead>
<tr>
<th>Rule</th>
<th>interest</th>
<th>unexpectedness</th>
</tr>
</thead>
<tbody>
<tr>
<td>{Q_{4T3:1.85/1.84} = 0, Type5 = 0}</td>
<td>8.56</td>
<td>7.37</td>
</tr>
<tr>
<td>{Q_{5T3:1.71/3.76} = 0, Type5 = 0}</td>
<td>8.56</td>
<td>7.10</td>
</tr>
<tr>
<td>{Q_{7T2:2.186/2.954} = 0, Type6 = 0}</td>
<td>7.41</td>
<td>6.46</td>
</tr>
<tr>
<td>{Q_{6T0.35/0.42} = 0, Type6 = 0}</td>
<td>5.37</td>
<td>4.59</td>
</tr>
</tbody>
</table>

... and Type2 in Table 7.8; actually there is no Type 1 result in the table because its unexpectedness is relatively lower.

Table 7.8: Mined unexpected dependence rules of Type 2 questions in subset of the data of \(Type1 = M \lor Type2 = M\).

<table>
<thead>
<tr>
<th>No.</th>
<th>Rule</th>
<th>interest</th>
<th>unexpectedness</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>{Q_{19T2:2.516/2.8325} = 0, Type6 = 0}</td>
<td>3.46</td>
<td>2.48</td>
</tr>
<tr>
<td>2</td>
<td>{Q_{26T6:0.35/0.42} = 0, Type2 = 0}</td>
<td>3.22</td>
<td>2.29</td>
</tr>
<tr>
<td>3</td>
<td>{Q_{20T2:7.942/7.63} = 0, Type6 = 0}</td>
<td>2.56</td>
<td>2.19</td>
</tr>
<tr>
<td>4</td>
<td>{Q_{21T4:4.452/4.45} = 0, Type2 = 0}</td>
<td>2.92</td>
<td>1.63</td>
</tr>
<tr>
<td>5</td>
<td>{Q_{16T2:5.62/5.736} = 0, Type6 = 0}</td>
<td>2.50</td>
<td>1.60</td>
</tr>
<tr>
<td>6</td>
<td>{Q_{24T4:3.2618/3.26} = 0, Type2 = 0}</td>
<td>2.33</td>
<td>1.43</td>
</tr>
<tr>
<td>7</td>
<td>{Q_{22T4:17.353/17.35} = 0, Type2 = 0}</td>
<td>2.46</td>
<td>1.37</td>
</tr>
<tr>
<td>9</td>
<td>{Q_{16T2:5.62/5.736} = 0, Q_{27T6:2.186/2.954} = 0}</td>
<td>2.13</td>
<td>1.32</td>
</tr>
</tbody>
</table>

The interest of the first rule in Table 7.8 is 3.46:

\[
\frac{P(Q_{19T2} = 0 \land Type6 = 0 \mid \mathcal{C})}{P(Q_{19T2} = 0 \mid \mathcal{C})P(Type6 = 0 \mid \mathcal{C})} = 3.46,
\]  

where \(\mathcal{C} = (Type1 = M \lor Type2 = M)\). This rule states that for students who are in the medium level of test type 1 or type 2, they are highly probable to get zero score in type 6 when they fail on question 19 and vice versa.

The experts gave the following comments. The first rule \(\{Q_{19T2:2.516/2.8325} = 0, Type6 = 0\} \lor \{Q_{20T2:7.942/7.63} = 0, Type6 = 0\} \lor \{Q_{24T4:3.2618/3.26} = 0, Type2 = 0\} \lor \{Q_{22T4:17.353/17.35} = 0, Type2 = 0\} \lor \{Q_{16T2:5.62/5.736} = 0, Q_{27T6:2.186/2.954} = 0\} \equiv \mathcal{C}\) confirms the missing link between questions and types.
0. $Type6 = 0$ is regarded as expected, because it belongs to the S3 class (reciprocal thinking or negative thinking) (Steinle and Stacey, 2003). Further, rule (11) $\{Q18_{T2.0.426/0.3} = 0, Type5 = 2\}$ seems surprising to them and they would like to see some explanations for it.

**Explanations**

The first explanation of the first rule is plotted in Figure 7.12. It shows that $Q20_{T2}$ and $Q28_{T6}$ are the possible factors for this rule. The experts thought the explanation is reasonable although they do not think of it in this way. However, they did not think it is useful because they already knew other reasons for this rule.

![Figure 7.12: Explanation for rule $\{Q19_{T2} = 0, Type6 = 0\}$.

The second explanation of rule (1) is plotted in Figure 7.13. $Type2$ and $Q7_{T1}$ are possible factors. The experts found the $Q7_{T1}$ node to be surprising because the questions of type 1 and type 2 are designed to be opposite. That is, $Q7_{T1}$ and $Q19_{T2}$ are supposed to have opposite response. However, $Q7_{T1} = 1 \land Q19_{T2} = 1$ is the major response of this subset of students (see Table 7.9); it contradicts the experts’ knowledge. Because the dataset is part of the unclassified students ($Type1 = M \lor Type2 = M$) that they are not familiar with, they wonder how the associations in other groups of students are.

![Figure 7.13: Another explanation for rule $\{Q19_{T2} = 0, Type6 = 0\}$.

}
Table 7.9: The contingency table of $Q_{19T2}$ and $Q_{7T1}$.

<table>
<thead>
<tr>
<th>$Q_{7T1}$</th>
<th>$Q_{19T2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>69</td>
</tr>
<tr>
<td>1</td>
<td>113</td>
</tr>
</tbody>
</table>

7.2.2 Analysis of Rule $\{Q_{18T2} = 0, Type5 = 2\}$

In the previous round of study, there was one rule outside of the experts’ knowledge: $\{Q_{18T2} = 0, Type5 = 2\}$, and the experts wanted to know more about it. The statistics of the two variables are listed below.

- Number of examples of $Q_{18T2} = 0$: 106 (38.97%).
- Number of examples of $Type5 = 2$: 27 (9.93%).
- Expected probability: $P(Q_{18T2} = 0)P(Type5 = 2) = 3.87\%$.
- Actual probability: $P(Q_{18T2} = 0, Type5 = 2) = 7.72\%$.
- Interest: $\frac{P(Q_{18=0,Type5=2})}{P(Q_{18=0})P(Type5=2)} = 1.99$.

Figure 7.14 shows a sieve diagram of attributes $Q_{18}$ and $Type5$. A sieve diagram shows the frequencies in a two-way contingency table in relation to expected frequencies under independence. The observed frequency in each cell in a contingency table is shown by the number of squares drawn in each rectangle. Hence, the difference between observed and expected frequency appears as the density of shading, using color to indicate whether the deviation from independence is positive (blue) or negative (red). Therefore the cell is blue when interest $> 1$ and is red when interest $< 1$. It shows that $Q_{18} = 1$ and $Type5 = 3$ has largest expected frequency. However, $\{Q_{18} = 1, Type5 = 3\}$ is not so unexpected compared to $\{Q_{18} = 0, Type5 = 2\}$ by the definition of unexpectedness. We can perceive from the diagram that when students are correct in answering $Q_{18}$ ($Q_{18T2} = 1$), they are likely to be all correct in Type 5 questions. On the other hand, when students get wrong in $Q_{18}$ ($Q_{18T2} = 0$), few of them can be all correct in Type 5 questions.

Explanations
In terms of explanations, we are looking for alternative Bayesian networks that can approximate the conditional probability of \( P(Type5 = 2|Q18 = 0) \):

\[
P(Type5 = 2|Q18 = 0) = \frac{P(Type5 = 2, Q18 = 0)}{P(Q18 = 0)} = \frac{0.0772}{0.3897} \approx 19.81\% \quad (7.2)
\]

The first explanation for this rule is plotted in Figure 7.15; \( Q3_{T5} \) and \( Q8_{T1} \) are possible factors for this rule. The experts thought this explanation is similar to the second explanation; therefore, they paid more attention to the next explanation.

\[
\begin{align*}
Q18_{T2} & \rightarrow Q3_{T5} \rightarrow Type5 \\
& \quad \quad Q8_{T1}
\end{align*}
\]

Figure 7.15: Possible explanation for the rule \( \{Q18 = 0, Type5 = 2\} \).

Figure 7.16 shows the second explanation; \( Q3_{T5} \) and \( Type2 \) are possible factors. Figure 7.17 and 7.18 show the sieve diagram of \( Q18, Q3 \) and \( Type5 \) respectively. We will use ‘positive dependence’ to describe blue rectangles in the sieve diagram for the positive deviation from independence. The rectangle (in Figure 7.17) of \( Q18 = 0, Q3 = 1 \) is red means the actual frequency is lower
then expected. On the other hand, The rectangle of $Q_{18} = 0, Q_3 = 0$ is blue, meaning the actual frequency is higher than expected. In other words, we can say that students tend to get $Q_3 = 0$ when they get $Q_{18} = 0$ comparing to the marginal distribution of $Q_3$.

For the second variable $Type_2$, its values 0,1,2 and 3 have a positive dependence with $Q_{18} = 0$ (Figure 7.19). Although it is not probabilistically sound, let us simplify the association that $Q_3 = 0$ and $Type_2 = \{0,1,2,3\}$ have higher tendency when $Q_{18} = 0$. In the next step, we can associate that $Type_5 = 2$ has higher tendency when $Type_2 = \{2,3\}$ (Figure 7.20). The rough inference shows why $P(Type_5 = 2|Q_{18} = 0)$ is higher than expected.

![Figure 7.16: Second explanation for the rule \{Q_{18} = 0, Type_5 = 2\}.](image)

![Figure 7.17: Sieve diagram of $Q_{18}$ and $Q_3$.](image)

![Figure 7.18: Sieve diagram of $Q_3$ and $Type_5$.](image)

The experts think that it makes sense why $Q_{3_{T5}}$ and $Type_2$ are related to the rule \{$Q_{18_{T2}} = 0, Type_5 = 2$\}. In both explanations, the pattern $Q_{18} \rightarrow Q_3 \rightarrow Type_5$ draws the experts’ attention to the possible factor of leading zero. Table 7.10 shows the question $Q_{18}$ and the three questions of Type 5;
CHAPTER 7. CASE STUDIES

Figure 7.19: Sieve diagram of $Q_{18}$ and $Q_3$.

Figure 7.20: Sieve diagram of $Type_2$ and $Type_5$.

since only $Q_3$ of Type 5 has leading zero, which is the same with $Q_{18}$, it could be the reason why $Type_5 = 2$. The explanation could be that students might have misconception in comparing decimal numbers with leading zero ($Q_{18}$ and $Q_3$); but the students can correctly answer $Q_4$ and $Q_5$.

Table 7.10: Items related to the rule \{ $Q_{18} T_2 = 0$, $Type_5 = 2$ \}

<table>
<thead>
<tr>
<th>Item Type</th>
<th>Item</th>
<th>Questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type 2</td>
<td>$Q_{18}$</td>
<td>0.426/0.3</td>
</tr>
<tr>
<td>Type 5</td>
<td>$Q_3$</td>
<td>0.3/0.4</td>
</tr>
<tr>
<td>Type 5</td>
<td>$Q_4$</td>
<td>1.85/1.84</td>
</tr>
<tr>
<td>Type 5</td>
<td>$Q_5$</td>
<td>3.71/3.76</td>
</tr>
</tbody>
</table>

Although the design of unexpectedness is iterative that the KB could be updated, in the DCT case study, the experts thought it is not necessary to do so; because they believe the original BN is the one mostly close to their knowledge.

7.3 Evaluation of Experiments

The evaluation of the case studies is based on the users’ experience of the proposed KDD design. We discuss it in three aspects – usability of knowledge acquisition, surprisingness of unexpected patterns, and reasonableness of explanations.
7.3.1 Usability of Knowledge Acquisition

The acquisition of the Bayesian network (served as KB) in the DCT case study is straightforward because the educational experts already had experience in modeling their knowledge in a BN in their previous research (Nicholson et al., 2001). Because the parameters of the network (i.e. the CPTs) are estimated directly from the frequencies in the data, there are no subjective probabilities required; thus, all that the experts need to provide is the structure of the BN.

In the case study of ANZDATA, whilst the expert confirmed that the interface is easy and effective to use, some confusion was encountered in assigning the association graph. We find the confusion is caused by three factors: (1) graphical semantics of BNs, (2) the arrangement of variables and values, and (3) the simplicity of the knowledge acquisition. Firstly, because the data analyst (the author) failed to effectively communicate with the medical expert, so that the expert generalised the meaning of ‘relates-to’, which originally implies ‘directly relates-to’, to a broader meaning of ‘anyhow relates-to’. Therefore, the expert thinks that it is not easy to assign every ‘anyhow relates-to’ associations; for example, the expert assigns $A \rightarrow B$ because $A \rightarrow x_1 \rightarrow x_1 \rightarrow \cdots \rightarrow B$.

The second factor that confuses the medical expert is the arrangement of some variables and values. Sometimes, a particular value in a variable is a distinct medical concept which would be unrelated to other possible values; it implies that two different values in a variable might be associated with different factors. Therefore, assigning associations in the level of variables (attributes) could be confusing in this case. For example, the expert explains that the attribute diseaseb (disease that causes end stage renal disease) consists of the values ‘diabetes’, ‘glomerulonephritis’, ‘polycystic kidney disease’, etc.. These diseases do not relate to the same risk-factors or effects; thus connecting the attribute diseaseb to other attributes could not precisely represent the expert’s knowledge.

Finally, although the knowledge acquisition is designed as simply as possible, its virtue might become its drawback from the user’s point of view. The medical expert complains that assigning the ‘relates-to’ associations is restrictive and limited in expressing her full knowledge. For example, normal people who have higher BMI (body mass index) values are likely to have diabetes, coronary artery disease and perhaps cancer; however, for dialysis patients these relationships are reversed. The example reflects the weakness of the design of
knowledge representation - that is, only the variables of the data are modeled in a BN; and representing the knowledge of the example would require an additional node to represent normal people or dialysis patients. Meanwhile, the design of ‘relates-to’ associations is incapable of acquiring detailed knowledge from experts; in the example, the expert wanted to input ‘higher’ BMI → Diabetes but found only ‘relates-to’ links could be assigned. It seems to be a dilemma in choosing between the simpleness and complexity of knowledge representation. When we asked the preference of the medical expert about using a more complex language for knowledge acquisition, simpler language was preferred instead, even though it is restrictive.

In terms of the simpleness/complexness dilemma, the expert suggested a desired function for easing the problem. It is desired to have an interface that can responsively provide value-level associations when the attribute-level association is assigned. Furthermore, it would be more useful if the system could find an ‘association-chain’. For example, if the user assigns Diabetes → TransplantList, it would be beneficial if the system could recover the association Diabetes=T1 → CoronaryArteryDisease=Yes → TransplantList=Yes from the data. Surprisingly, what the expert suggested is in fact what the proposed explanation generation is designed for.

In addition to the aforementioned problems, we had also noticed a problem that would cause inaccuracy of KB. The problem arises from the user’s misinterpretation of the name of the attribute, so that the user thought that what was input into KB is what she meant. For example, the medical expert in mining ANZDATA was not familiar with the alignment of the encoding of names and values of attributes and the expert’s internal knowledge; thus, the user thought the attribute dryweight means the variable (changing) weight of patients; actually, dryweight means the absolute (static) weight. Therefore, the connections of dryweight were assigned in the sense of another meaning. We call the problem an inaccuracy problem of knowledge acquisition.

To sum up, the design of knowledge acquisition is proved to be applicable to real-world data no matter the user’s familiarity with BNs. However, the precise meaning of ‘relates-to’ associations needs to be fully explained to the user so that indirect associations will not bother the user. Second, the content of an attribute could sometimes confuse the user in assigning related variables because of the distinct meaning and associations of different values; this problem could be solved by rearranging the values into new attributes.
Thirdly, there is a simpleness/complexness dilemma observed - users want to express knowledge in detail but are not willing to spend more time in writing it down. Accordingly, the expert has suggested a direction which automatically completes the details of associations. Finally, the inaccuracy problem needs to be avoided by making sure that the user understands the context of attributes correctly; this problem could be prevented by providing the summaries of variables.

### 7.3.2 Surprisingness of Unexpected Patterns

The good news in the case studies is that surprising rules have been successfully discovered. In ANZDATA, the mined rules about a patient’s status on the transplant list conflict with the user’s domain knowledge (Rules 5,6 in Table 7.3). In DCT, the top two unexpected rules in the first trial (Table 7.5) rediscovered the set of students who are not understood by the users. Furthermore, the investigation into the item level (each question) also discovered a questionable rule which is out of the users’ knowledge.

Meanwhile, from the deep discussion with the users (domain experts), we have also learned more about why a rule could be unsurprising. If a rule is unsurprising, there might be three reasons for it:

1. The user knows more than the KB does; in the situation, some of the DK is not captured in the KB. This case has been discussed in Subsection 6.3.1. We call it an ‘uncaptured’ problem in later discussion.

2. Although indirect associations of the rule are modeled in the KB, the calculated unexpectedness is still high due to the inference of BNs. For example, the user might have input $A \rightarrow B \rightarrow C \rightarrow D$ into the BN, but the inference of the conditional probability $P(D|A)$ of rule $A \rightarrow D$ is still not precise enough. We call it an ‘imprecise’ problem in later discussion.

3. The context, format or values of the rule are not informative enough for the user to decide surprisingness. We call it an ‘uninformative’ problem in later discussion.

In terms of the uncaptured problem, the rules with Year of DCT (Rules 3 to 5 of Table 7.5) are unsurprising because of it. In Figure 7.5, the Year node has no connection with other nodes. However, the users already knew
the associations of Year to other variables; but they did not explicitly assign the connections because they intended to model them in another way.

There are many examples of the imprecise problem. In DCT, the Rules 1,3,5 in Table 7.8 are ranked as unexpected because the BN could not precisely model their joint probability. For example, although the BN has the connections $Q_{19} \rightarrow Type_2 \rightarrow FineCode \rightarrow Type_6$, the inference of $\hat{P}(Q_{19} | Type_2, Type_6)$ is still very different from the actual frequency in data.

Finally, the uninformative problem is usually the reason behind the comment of ‘meaningless’ which makes the user unable to decide surprisingness. For Rules 1 to 4 of ANZDATA in Table 7.3, the user commented on various reasons for why the rules are unsurprising, and the reasons generally have the uninformative problem. We try to analyse the uninformative problem in depth in Subsection 7.3.4.

### 7.3.3 Utility of Explanations

The two explanations in Figures 7.3 and 7.4 of ANZDATA are regarded as reasonable because the user conjectures that the late referral factor implies more complications and a higher risk of infection which may delay the patients’ opportunity of waiting for a transplant. In the DCT data, the explanations in Figures 7.15 and 7.16 shed light on the factor of leading-zero which is believed to be the main reason for the rule under examination.

There are occasions when users think the explanation is reasonable, but it is not the actual explanation in their mind. For example, the users regarded the explanation in Figure 7.12 as reasonable because they knew that the intermediate factors were somehow associated with the variables $Q_{19} | Type_2$ and $Type_6$; however, they thought it was not the actual explanation for the rule.

Finally, some explanations did not provide any help to the users; furthermore, some explanations surprised and confused the users. For example, the educational experts thought the explanations in Figure 7.8 were not helpful because they already knew how Year is associated with FineCode, but the reasons are not in the dataset. The explanation in Figure 7.13 surprised and confused the users because the factor $Q_{7} | T_1$ of $Type_1$ items should have an opposite distribution to the attribute $Q_{19} | T_2$ of the rule.
7.3.4 Identified Gaps in Mining Surprising Patterns

Whilst the results of case studies are encouraging, we have identified more gaps of uninformative problems. In the case study of ANZDATA, there are two types of identified gaps: literal gaps and reasoning gaps. The literal gaps happen when the data is encoded differently to how the user expects. For example, rule 1 consists of attribute \textit{dryweight} in the range between 53.2 – 101.5 kg; however, the expert is used to reasoning a patient’s weight as overweight or underweight. Secondly, the reasoning gaps are caused by the difference between a user’s reasoning and the nature of association rules. In the following paragraph, we discuss several reasons for literal and reasoning gaps by giving examples from the user’s interpretation.

\textit{Reasons for literal gaps}

\textbf{Improper discretisation} - Continuous data is not discretised into meaningful range. The expert thinks \textit{dryweight} = 53.2 - 101.5 kg of rules 1 to 3 is not meaningful because adults usually have a weight in that range.

\textbf{Temporal information loss} - The case is due to information loss when summarizing multiple records. The expert thinks the change of \textit{dryweight} is more meaningful than an average value; that is, gaining or losing weight is the way the expert reasons. The values of \textit{txwlcat} (transplant waiting list) do not provide sufficient information because the expert also needs to know the change of status of a transplant list.

\textbf{Exhaustive and mutually exclusive property of variables} - If a variable is exhaustive, all possible values are enumerated. If the values of variables are mutually exclusive, they have no overlapping meanings. In the result, value \textit{mideath} = \textit{AliveorNon-MIdeath} of rule 2 is not exclusive because the value has an overlapping meaning – alive or non-myocardial infarction death.

\textbf{Semantic gap} - The encoding of data is not consistent with the user’s knowledge in the semantic aspect. For example, \textit{dryweight} is the \textit{absolute} (static) weight of a patient; however, the expert’s knowledge is based on \textit{relative} (changing) weight on a per patient basis.

\textit{Reasoning} gaps are caused by the difference between a user’s reasoning and the nature of association rules.
### Table 7.11: The alignments between the identified gaps and the five steps of KDD.

<table>
<thead>
<tr>
<th>Gap</th>
<th>Sel.</th>
<th>Prep.</th>
<th>Trans.</th>
<th>Mining</th>
<th>Inter.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Improper discretisation</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Temporal information loss</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exhaustive and mutually exclusive property of variables</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Semantic gap</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Partial alignment of mined rules and experts’ knowledge</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Cause-effect context</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

**Reasons for reasoning gaps**

**Partial alignment of mined rules and expert’s knowledge** - An association rule covers only a subset of grouped factors in an expert’s knowledge. In rules 1 to 3, the `dryweight` attribute is the case, because weight is subject to sex and age, but they are not shown in the rule.

**Cause-effect context** - Association rules do not have the context of causality; however, users usually interpret rules in the context of cause and effect. Rule 4 is a case of this reason; the expert points out that the status of the transplant list is not decided by cause of death, because the attribute of cause of death is on the left hand side.

On account of *literal* and *reasoning* gaps, the expert felt that the rules were not *meaningful* enough for gaining new knowledge, and therefore categorised rules 1 to 4 as ‘meaningless’. Actually, the various reasons in these gaps correspond to different stages in the KDD process. In the schema of Fayyad et al. (1996), there are five steps in a KDD cycle - namely data *selection*, *preprocessing*, *transformation*, *mining*, and *interpretation*. Table 7.11 lists the alignments between the identified gaps and the five steps of KDD.
7.4 Summary

This chapter reports two case studies used to evaluate the generation of surprising patterns. Encouragingly, some surprising rules were discovered within a very small set of unexpected rules. Unfortunately, none of the results could draw the experts’ interest to initiate new research; nevertheless, this wishful expectation is out of the scope of the research question. In terms of explanations, we also find some positive feedback that some associations that the experts were previously unaware of are triggered by the generated explanations. Finally, the case studies are evaluated and discussed in three aspects in Section 7.3 – usability of knowledge acquisition, surprisingness of unexpected patterns, and reasonableness of explanations. More gaps learned from the case studies are reported in Subsection 7.3.4.
Chapter 8

Discussion and Conclusion

This thesis presented a study of mining surprising patterns from a different perspective to earlier work on unexpectedness. Previous research on unexpectedness approached the task of mining surprising patterns primarily as an exercise in rule ranking, by using or defining a particular representation of the user’s knowledge and using a variety of functions to rank rules, with the assumption that rules not supported in the knowledgebase are ‘surprising’. However, this viewpoint does not pragmatically take a user’s opinion into consideration. Determining whether a pattern is surprising (so that the pattern might be interesting or useful) is not a trivial task to the user. A number of factors may potentially hinder the user from correctly interpreting patterns, such as misunderstanding the context of rules, insufficient information provided during pattern interpretation, and feelings that the mined rules are meaningless/pointless.

Further, the requirement of a knowledgebase in unexpectedness causes another well-known problem in knowledge engineering: the knowledge acquisition bottleneck (Subsection 2.6.1). Complex knowledge representation incurs high overhead to acquire knowledge from a user, because a knowledge engineer (or, even more problematically, the data analyst) has to articulate the language of the representation, and the user needs to understand the basics of the representation as well. Moreover, domain experts, who are likely to be the KDD user, are usually busy people (Jones, 1989); therefore shorter acquisition time is better in terms of usability. Several studies have tackled the problem via novel approaches, such as the learning of user’s prior knowledge from interactive feedback (Xin et al., 2006). Nevertheless, the usability problem is rarely formally addressed in the literature.

This thesis extends the scope of unexpectedness to knowledge acquisition,
meaningfulness and interpretation assistance. Chapter 2, presented an analysis of various knowledge representations that have been used in earlier work on unexpectedness (Subsection 2.4.5); on the basis of that analysis, Bayesian Networks (BN) were selected as the most appropriate platform of knowledge representation because of capabilities in encoding pairwise relations between variables, and collectively using these relations to make inference. New building blocks for BN based knowledge acquisition were presented in Chapter 5; they are (1) a graph converting algorithm (Subsection 5.3.3) for translating user assigned cyclic graphs (association graphs) to DAGs for BNs, and (2) the use of a statistical independence test for association rules (Subsection 5.3.2) to ensure the manually updated edges of BNs represent correlations. For dealing with large datasets (with numerous variables), the local-BN generating algorithm was proposed (Subsection 5.3.4) to address scalability problems when utilising large BNs.

These new building blocks were combined with new metrics for unexpectedness into an architecture that can iteratively acquire-apply-update the KB to mine unexpected association/dependence rules (Subsection 5.3.6). A case study on Danish geriatrics (in Subsection 5.4.1) demonstrated the effectiveness of the architecture; notably, the proposed design propagated the associated KB at a faster pace, and unexpected rules were mined in earlier iterations in comparison with mining unexpected frequent itemsets (Jaroszewicz and Simovici, 2004).

The iterative design of knowledge acquisition was evaluated by conducting comprehensive data mining experiments on real-world data with users who are domain experts (Chapter 7). These experiments confirmed that the knowledge acquisition interface is relatively easy and effective to use, even with little prior knowledge about Bayesian networks; however, some precautions need to be taken before applying the design generally, as discussed in Subsection 7.3.1.

The meaningless problem of mined rules was discussed in Chapter 4; it is a problem that hinders the interpretation of rules’ surprisingness. Possible types of meaningless rules are: trivial, uninformative, or nonsensical. Whilst the actual mechanism of a user categorising a rule as meaningless is not clear, an approach to restricting the search space of association rules according to user-specified semantic structures was devised, in order to avoid meaningless rules (Section 4.1). The proposed method was evaluated on a large medical dataset (ANZDATA), and passed the evaluation criterion that expert-expected
associations have been discovered within a small set of rules (Section 4.1.2). However, an exploration of more comprehensive case studies (Subsection 7.3.4) indicated that there are more (and more complex) causes of meaningless rules which are seemingly beyond the capability of the proposed ontological approach.

In an attempt to reduce the effort required for knowledge acquisition, pre-established domain ontologies were explored as the KB for unexpectedness. Section 4.2 outlined a hypothetical design to build a Bayesian network from a domain ontology, where the BN would then be used as the KB. An important conclusion was that individual-level (attribute-level) relations are the essentials for applying ontology knowledgebases to mining unexpected patterns via this design. In fact, there are few openly accessible domain ontology bases having individual-level relations; thus the hypothetical design is impractical for now.

During the early stages of the case study analyses detailed in this thesis, it was observed that, when forming a definite judgement on a rule’s surprisingness, domain experts perform complex reasoning by considering all relevant factors and their relations together (Chapter 6). On the assumption that an expert would explain rules based on their knowledge, it was expected that if a rule could not be explained, then it should be surprising; but in reality a common situation was that the expert believed the existence of possible explanations for the rule but could not recall them instantly. In an effort to simplify the process of rule assessment, it was proposed that generating possible explanations from the data would prove helpful in assisting pattern interpretation.

Chapter 6 presents a novel approach to generating possible explanations for mined rules from data. The approach is based on modelling reasoning chains by comparing the inferred conditional probability with the actual conditional probability of a rule. The reasoning chains of explanations are modeled in Bayesian networks in order to be generalised to more complex structures, and presented to the data mining user as an aid to understanding and explaining a rule. The case studies detailed in Section 6.5 demonstrated that this approach is capable of generating reasonable explanations from data effectively; and can provide alternative suggestions to update the BN in mining unexpected patterns.

Surprising patterns have been successfully discovered in two domains using users who are domain experts Chapter 7. The first case study, based on a comprehensive database recording progression and treatment of kidney disease in
many patients (ANZDATA), demonstrated that 33.3% of the top 6 unexpected rules in the first iteration were regarded as surprising by the nephrologist (Section 7.1). 40% of the top 5 unexpected rules of the second case study (in the education domain – DCT dataset) were regarded as interesting. The experts did not think the rules were surprising, as they already knew of their existence, but the rules were inexplicable to the experts - that is, the experts understood that the particular relationship expressed in the rule existed in the data, but did not know why, despite being extremely knowledgable about the domain (Section 7.2). The following DM trial, on a subset of the education data focusing on the ‘interesting’ rules, found one surprising rule out of 11 unexpected rules (Subsection 7.2.1). A detailed discussion of the reasons for why unexpected rules might not be surprising appears in Subsection 7.3.2.

Many of the generated explanations in the case studies are reasonable to the experts and helped the experts recall relevant factors to explain the rules. Subsection 7.3.3 reported the utility of ‘explanations’ from the users’ viewpoint.

Finally, the case studies presented in this thesis have identified more gaps in mining surprising patterns. Broadly speaking, there are two types of gaps: literal gaps and reasoning gaps. Literal gaps arise when a user’s interpretation of the variable names and data encoding in the dataset differs from the actual data encoding. Reasoning gaps are a consequence of the difference between a user’s reasoning and the nature of association rules. These categories of gaps could be generalised to uninformative problems, in which the rule does not provide sufficient, intuitively correct or complete information to the user. Further discussion of how these gaps open the door to future research is in Section 8.3.

8.1 Revisiting the Research Question

In revisiting the ‘mining surprising patterns’ question, this thesis argues that the essentials are:

- Acquire a knowledgebase from the user.
- Define the measure of unexpectedness.
- Assist the user in pattern interpretation.
• Ensure the rules are mined properly (so that the rules are meaningful to the user).

In the research, the first two points are formally addressed in Chapter 5, and the overall design is proved to be capable of mining surprising patterns in real-world data when the users are domain experts (Chapter 7).

This thesis argues that it is important to assist the user in interpreting unexpected rules, and therefore proposes a new approach to explanation generation. The case studies have shown that some generated explanations can remind the users of factors they were previously unaware of when considering the data mining results. In terms of knowledge aggregation, presenting explanations to the user would help the user update the KB in a useful and consistent manner.

The research also suggests the existence of a prerequisite to the surprisingness problem; here termed the meaningfulness prerequisite – if a rule is meaningless to the user, the rule would not be considered as surprising. In the analyses (Chapter 4 and Subsection 7.3.4), the major cause of meaningless rules is that the rules do not provide sufficient information to the user (the uninformative problem). The first design of ontology-driven data mining in Chapter 4 attempted to address the meaningless problem by restricting the search space of association rules mining. In retrospect, it can prevent pointless or meaningless rules due to duplicated or mutually exclusive variables; moreover, it is capable of recovering meaningful and already known knowledge as demonstrated with the ANZDATA case study. However, the ontology approach could not prevent other kinds of uninformative problems.

The uninformative problem seems to be caused most commonly by improper data preparation\(^1\) and the natural characteristic of rules. The problem is further discussed as literal and reasoning gaps in Subsection 7.3.4. A discussion of the subtypes of uninformative problem (literal and reasoning gaps) indicates the importance of other steps of KDD\(^2\) prior to the scope of unexpectedness.

\(^1\)Including data selection, preprocessing and transformation.

\(^2\)That is, the steps of data selection, preprocessing, transformation and mining.
8.2. APPLICABILITY AND LIMITATIONS

To summarise, this thesis has presented a pragmatic design and new building blocks for the mining of surprising patterns, and demonstrated the value of the design in a set of comprehensive case studies. Extensive analysis of the case studies has suggested that the scope of surprisingness should be extended to the stages of data preparation and rules mining in order to present meaningful and surprising rules to users.

8.2 Applicability and Limitations

The solutions proposed in this thesis may be applied to the KDD of generic rules mining as long as two conditions are satisfied: (1) the dataset is tabular and non-temporal, and (2) the user can provide domain knowledge in the format of ‘relates-to’ associations. There are two reasons for why the solutions are incapable of handling temporal data. First, the design does not incorporate DM algorithms which deal the temporal dimension; a comprehensive survey of temporal DM may be found in (Roddick and Spiliopoulou, 2002). Secondly, the knowledge representation in the design, Bayesian networks, can process non-temporal data only; otherwise, ‘Dynamic Bayesian Networks’ would be an appropriate approach for temporal data (Burns and Morrison, 2003).

The design of unexpectedness is subject to the following issues. First, it is subject to the KDD stages prior to data mining, i.e. data selection, pre-processing and transformation; if the data is not properly prepared, the user may find difficulties in pattern interpretation. Secondly, it does not encompass the issues caused by the nature of rules mining; for example, the user may require temporal information or causality from conventional association rules. In terms of knowledge acquisition, there is a dilemma in the choice of simpleness/complexness; the user might require richer knowledge representation but might be unwilling to spend time on writing down DK. Finally, if the user is not familiar with the data or the semantics of rules, the user might misinterpret the rules.

The approach to explanation generation is restricted in three ways. First, it cannot detect ‘direct’ association between variables because it assumes intermediate factor(s) is(are) involved. Secondly, only observed variables in the data are considered as the candidate factors for explanations; thus, any unobserved factor is not considered. Finally, it cannot determine cause-effect relations; instead, only associations can be uncovered.
In terms of users’ domain knowledge, the approach presented in this thesis is applicable to any degree of acquaintance of the problem domain – from naïve users to domain experts – for the discovering of surprising patterns. However, there are two prerequisites for applying the proposed design. First, the data analyst, who operates the KDD system, needs some amount of domain knowledge so that the data can be prepared/processed correctly; the knowledge might come from the user when the user is a domain expert, or additional assistance from a third party may be needed. Secondly, the user needs to be familiar with the alignment of the encoding of names and values of attributes, so that the inaccuracy problem of knowledge acquisition could be avoided; therefore, presenting a summary of attributes to the user is recommended.

8.3 Future Research

This thesis has presented an initial exploration of aspects of mining surprising patterns. The new building blocks and approaches presented exhibit promising behaviour in real-world KDD projects. Of course, the design of the combination of the building blocks is not necessary the best way of using them, and it is recommended that they should be tailored to fit into individual KDD projects. For example, if there is an available domain ontology, then the element of ontology-driven DM (Section 4.1) could be added into the design. In the following subsections, future research is discussed in four aspects: (a) the source of knowledge, (b) the property of data, (c) the quality of rules, and (d) the presentation of ‘explanations’.

8.3.1 Sources of Knowledge

In Chapter 4, the possibility of using domain ontologies as the source of KB was investigated. In the presented design of unexpectedness (Chapter 5), the user and data are the sources of KB, where the user assigns the structure of a BN based on the data and the parameters of the BN are learned from data.

Early in the development of this research, domain literature was considered as a possible source of KB. If the KB is acquired from domain literature, the mined patterns should be ‘surprising’ to the domain rather than a particular user. The idea is challenging because it is likely that sophisticated natural language processing would be required to translate the knowledge into the KB;
moreover, it is not clear what representation could capture the complex concepts in the literature. Acquiring knowledge from the literature, nevertheless, would be a powerful technique not only in surprisingness but also other applications. The idea has been studied by Antal et al. (2004) where information from free-text resources is integrated with data in learning Bayesian networks for building clinical models of ovarian tumors.

The design for knowledge acquisition, extended from the work by Jaroszewicz and Simovici (2004), made an assumption that the probability distribution of data is similar to that of the user’s domain knowledge. However, this assumption might be invalid in some situations. If the probability distribution of data was very different to the user’s knowledge, then learning the parameters of the underlying BN from the data will certainly miss surprising rules. A hypothetical example is that in mining a dataset of lung cancer, a user might associate smoking with cancer in the KB assuming that smoking causes cancer. However, if not smoking is more prevalent in cancer patients, supported by the dataset, the KDD system will miss this contradictory fact because the parameters of BN are learned from data. An intuitive solution is to use another dataset, which was well studied by the user, as the source of KB. The idea might be practical in the domains where data is kept up-to-date through periodic or occasional addition and deletion, e.g. evolving data (Ganti et al., 2001).

The identified ‘literal gaps’ (Subsection 7.3.4) cause inaccuracy in knowledge acquisition. In the early stages of this research, a user (also an expert) was asked to assign associations according to the names of attributes; however, during pattern interpretation the expert found that what was encoded in the data did not match their original interpretation of that attribute. For example, the expert assigned dryweight relates to bmi; and later realised that their conceptualisation of ‘dryweight’ was not the actual data recorded for that attribute. As a consequence, the acquired KB contained misinterpretations due to ‘literal gaps’. In addressing this problem, a strong-rule assumption, where strong rules should be the common sense to domain experts (assumption of exception rules, see Subsection 2.3.1), could be applied when the users are domain experts. A theoretical design of mining surprising patterns using this assumption is outlined in Appendix F.

Finally, computational assistance for knowledge acquisition is also worthy of research. As discussed in Subsection 7.3.1, the expert suggested that
it would be beneficial if the system can automatically generate ‘association-chains’, e.g. $A \rightarrow X_1 \rightarrow X_2 \rightarrow \cdots \rightarrow B$, when an association $A \rightarrow B$ is input; clearly, the method of explanation generation described here may be applied to this task. Further, if the interface can responsively provide value-level associations during interaction, any inconsistency between DK and data (as aforementioned in the discussing of using data as the source of KB) are likely to be noticed by the user.

\subsection{Data Property}

The thesis deals with only tabular and non-temporal data. Future research could replace the DM algorithm and the representation of KB to handle other types of data. For example, temporal DM algorithms and dynamic Bayesian networks could be applied to various types of temporal data (Roddick and Spiliopoulou, 2002; Burns and Morrison, 2003; Winarko and Roddick, 2005; Mallaug and Bratbergsengen, 2005; Stacey and McGregor, 2007).

For the scenario of dynamic database environments, keeping patterns up-to-date and finding ‘patterns of evolving patterns’ are important challenges in the future (Kriegel et al., 2007). Time-evolving data and data streams may be related subject matter for extending the scope of mining surprising patterns (Ganti et al., 2001; Cormode and Hadjieleftheriou, 2008).

\subsection{Rule Quality}

Corresponding to the problem of meaningless rules, earlier KDD literature has stressed the importance of variable selection (feature selection) (Liu and Motoda, 1998; Guyon and Elisseeff, 2003), data preprocessing and transformation (Han and Kamber, 2006, chap. 2). Incorporating earlier techniques of feature selection, data preprocessing and transformation into the framework of surprisingness is an obvious candidate for further research.

In addressing the ‘literal gaps’ (Subsection 7.3.4), a systematic approach is desirable to use domain knowledge (acquired from domain experts or other sources) to transform the feature space into more meaningful representations. For example, numerical attributes must be discretised meaningfully and the values of attributes are mutually exclusive and exhaustive.

New theoretical development of ‘knowledge-rich’ association rule mining is also desirable to address ‘reasoning gaps’ (Subsection 7.3.4). Toward this
8.3. **FUTURE RESEARCH**

direction, the KDD system should be able to ‘reason’ whether a rule provides ‘enough’ information to the user and then proactively supplies the ‘missing’ information. For example, the system might be able to ‘reason’ the variable \textit{dryweight} (in Rules 1 to 3 in Table 7.3) alone is insufficient for a medical expert to characterise a particular group of patients, then the system automatically analyses the rule with respect to the factors \textit{age}, \textit{gender} and \textit{height}. Further, association rules with causal context is of interest as well, e.g. Liang et al. (2007).

Temporal information loss is another cause of meaningless rules. Upgrading the mining algorithm to incorporate temporal DM (Roddick and Spiliopoulou, 2002) might be a straightforward approach to this problem. In fact, the medical dataset used in the ANZDATA case study contains temporal information in the format of sequential records, and much of the expert’s knowledge is time-oriented. Therefore, extending the research to handling temporal data is critical to broader applications in the medical domain.

### 8.3.4 Presentation of Explanations

Presenting ‘explanations’ as BNs is sometimes inadequate to convey the implications of explanations to users. In the case studies presented here (Chapter 7), other presentation techniques were included – CPTs, sieve diagrams and histograms – to show the context of explanations. Future research can consider more techniques such as mosaic displays (Friendly, 2001) or visualization of graphical probabilistic models, e.g. bar charts or pie charts as demonstrated in Netica\(^3\) and GeNIe& Smile\(^4\), to present ‘explanations’.

### 8.3.5 Summary

Future research is required to improve the techniques presented in this thesis, particularly with regard to efficiency and accuracy (of identifying surprising patterns), in four aspects: (a) the source of knowledge, (b) the property of data, (c) the quality of rules, and (d) the presentation of ‘explanations’. Each

\(^3\)http://www.norsys.com/netica.html

\(^4\)http://genie.sis.pitt.edu/
aspect consists of many possibilities for new directions, but it should be noted that whilst adding new developments can resolve some problems, the KDD system would become more complex and less manageable.

8.4 Conclusion

In addressing the research question, the presented unexpectedness design is proved to be capable of mining surprising patterns. The essentials of mining surprising patterns are:

- A knowledgebase (KB) that captures the user’s domain knowledge (DK), where the proposed design utilises Bayesian networks as the knowledge representation, and the methods for knowledge acquisition allow the user to easily provide knowledge by simply assigning ‘relates-to’ associations.

- A measure for unexpectedness, defined as the difference between confidence and the inferred conditional probability for association rules; and the difference between interest and the inferred interest for dependence rules.

- Assistance for pattern interpretation, where explanations are found to be useful to help users reason rules better; therefore, a probabilistic approach to finding explanations was developed.

- Prevention of meaningless rules; if a rule is meaningless to the user, the user have difficulty in interpreting the rule; this may be partially resolved by ontology-driven data mining, but there are also other kinds of meaningless problems, e.g. uninformative problems, which need further research.

In conclusion, this thesis argues that the scope of mining surprising patterns should be extended to all stages of knowledge discovery and data mining (KDD), and the data mining algorithm should be advanced beyond conventional association/dependence rules mining to fully address the identified literal and reasoning gaps.
Bibliography


A. Srinivasan and R. D. King. Feature construction with inductive logic programming: A study of quantitative predictions of biological activity aided


Index

Bayesian networks, 197

Conflicting, 19
Contradiction, 19
  Contradictory, 19

Interestingness, 1, 17
  Interesting, 1, 18
  Objective, 1, 17
  Subjective, 1, 16–18

Knowledge
  Acquisition, 7, 52
  DK, 6, 18
  Domain knowledge, 6, 18
  KB, 6, 18
  Knowledgebase, 6, 18
  Representation, 26

Surprisingness, 2
  Surprising, 2, 20

Unexpectedness, 2, 6, 19, 25
  Unexpected, 2, 6, 19
Unknown, 20
Unrecognised, 20
Appendix A

Acronyms

AG: Association graph
AR: Association rule
BN: Bayesian network
CPT: Conditional probability table
DAG: Directed acyclic graph
DM: Data mining
DK: User’s domain knowledge
DR: Dependence rule
ER: Exception rule
KB: Knowledgebase
KDD: Knowledge discovery and data mining
LHS: Left hand side
RHS: Right hand side
Appendix B

Association Rules and Dependence Rules

In the thesis, the mining of unexpected patterns are based on association rules and dependence rules mining; their definitions and significance measures are introduced here. In earlier research, association rules are defined over binary attributes mainly from databases of customer transactions (basket data), e.g., in (Agrawal et al., 1993). Later, association rules are also generalised to categorical attributes where the basic unit, an item, is defined as an attribute-value pair.

Let \( \{A, v\} \) be an attribute-value pair of attribute=\( A \) and value=\( v \); and an item \( I \) is an attribute-value pair: \( I = \{A, v\} \). A set of items, e.g. \( \{I_1, I_2, \ldots, I_k\} \), is called an itemset. Then the frequent itemset is defined as:

**Definition 10** (Frequent Itemset). We say an itemset \( \mathcal{I} = \{I_1, I_2, \ldots, I_k\} \) is a frequent itemset if the joint probability of the items \( I_1 \) to \( I_k \) is greater or equal than a threshold:

\[
P(I_1, I_2, \ldots, I_k) \geq \text{minsupp}.
\]

We call the joint probability \( P(\mathcal{I}) \) of the itemset as **support**.

For example, \( \mathcal{I} = \{Diabetes = Type1, Cancer = NO\} \) is an itemset, and if the support of \( \mathcal{I} \) is greater than the adjustable threshold \( \text{minsupp} \), then \( \mathcal{I} \) is a frequent itemset. Based on itemsets, association rules are defined as:

**Definition 11** (Association Rules). An association rule is composed of an itemset on the left-hand-side and another item on the right-hand-side; let \( \mathcal{A} \)
and B be two distinct itemsets/item, its rule is written in the format \( A \rightarrow B \) which satisfies:

\[
P(B|A) \geq \text{minconf};
\]

\[
P(B, A) \geq \text{minsupp}.
\]

We call the conditional probability \( P(B|A) \) of the rule as \textit{confidence}.

For example,

\[
\{ \text{Diabetes} = \text{Type1}, \text{Cancer} = \text{NO} \} \rightarrow \text{Smoke} = \text{Never}
\]

is a rule where \( A = \{ \text{Diabetes} = \text{Type1}, \text{Cancer} = \text{NO} \} \) and \( B = \{ \text{Smoke} = \text{Never} \} \).

Although the support-confidence framework of association rules is widely accepted in many real-world data mining applications, such as basket data analysis, fraud detection, recommendation systems, some criticise that stating an association rule by itself, is at best incomplete information and at worst misleading (Silverstein et al., 1998).

\textbf{Example 5 (A Critique of Association Rules).} Suppose we have market basket data from a grocery store. Let us focus on the purchase of tea and coffee; as the table shows:

\[
\begin{array}{|c|c|c|}
\hline
\text{} & \text{Tea} & \text{¬Tea} \\
\hline
\text{Coffee} & 80 & 20 \\
\text{¬Coffee} & 40 & 10 \\
\hline
\end{array}
\]

Obviously, we can see an association rule

\[
\text{Coffee} \rightarrow \text{Tea}, \text{support} = 0.53, \text{confidence} = 0.8
\]

At this point, the store manager might decide to target coffee drinkers in his tea displays. However, the fact is that non-coffee drinkers also have the same probability of buying tea; actually, the presence of tea and coffee is independent.

\textbf{Problem 2 (Independent Association).} The edges/links of Bayesian networks represent dependent relationships; however if we consider association rules as suggestions for adding new links as Jaroszewicz and Simovici (2004) proposed,
chances are many independent links would be added. Here is a simple example demonstrating that association rules could sometimes be independent. Suppose a survey result of drink preference in the following table.

<table>
<thead>
<tr>
<th></th>
<th>Tea</th>
<th>¬Tea</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coffee</td>
<td>79</td>
<td>21</td>
</tr>
<tr>
<td>¬Coffee</td>
<td>41</td>
<td>9</td>
</tr>
</tbody>
</table>

¬Coffee → Tea is one of the association rule with confidence = 0.82. It says that people who dislike coffee usually prefer tea. Can we assign a link in the Bayesian network in between Coffee and Tea? Maybe not; the table also shows that people who like coffee usually be fond of tea as well. In fact, the variables Coffee and Tea are actually independent; we can perceive the fact from

\[ P(\text{Tea}|\text{Coffee}) \simeq P(\text{Tea}|\neg\text{Coffee}). \]

In addressing the problem of association rules, Silverstein et al. (1998) presents the concept of dependence rules that ensuring rules are statistically dependent. Statistically speaking, two events, e.g. \( x \) and \( y \), are independent if and only if \( P(x \land y) = P(x)P(y) \). If \( x \) and \( y \) are not independent, then we can a measure to indicate their dependence:

**Definition 12** (Measures of Interest). The interest measure of two events \( x \) and \( y \) is defined

\[
I_{\text{rule}}(xy) = \frac{P(xy)}{P(x)P(y)},
\]

wit the obvious extension to more than two events.

The dependence rules are defined as:

**Definition 13** (Dependence Rules). Two variables \( A \) and \( B \) are independent if they are tested statistically to be independent, such as using the \( \chi^2 \) test; otherwise they are dependent. Then if the interest \( I(a_i,b_j) \) of the itemset \( \{A = a_i, B = b_j\} \) is greater than a threshold: \( I_{\text{rule}}(a_i,b_j) \geq \epsilon_t \), we write \( \{A, B\} \) (as a short hand of \( \{A = a_i, B = b_j\} \)) as a dependence rule.
**B.1. Test for Independence for Association/Dependence Rules**

The purpose of testing rules’ independence is to identify rules with attributes that are truly dependent. Proposed in dependence rules (Silverstein et al., 1998), chi-squared ($\chi^2$) test could be applied to the test of independence for mined rules. Given an association rule $A \rightarrow B$, we can build a 2-way contingency table of $A, B$ as Table B.1 shows.

The expectation of each cell is calculated as

$$E_{i,j} = \frac{\sum_{i'} r_{i',j} \sum_{j'} r_{i,j'}}{N}.$$  \hspace{1cm} (B.2)

Then the $\chi^2$ value for attributes $A, B$ is calculated by

$$\chi^2 = \sum_{i,j} \frac{(r_{i,j} - E_{i,j})^2}{E_{i,j}}.$$  \hspace{1cm} (B.3)

When $\chi^2$ is near zero, it means the distribution in data is nearly a normal distribution; in other words, attributes $A$ and $B$ are nearly independent. On the other hand, if $\chi^2 > \chi^2_{\alpha,df}$, $A$ and $B$ are likely to be dependent. $\chi^2_{\alpha,df}$ is the quantile of $\chi^2$ distribution at $\alpha$ significance level and $df$ degree of freedom. The $\alpha$ is usually set at 0.05 or 0.01 and the $df$ is determined by the sizes of rows and columns in a table: $df = (\text{Rows} - 1)(\text{Columns} - 1)$. For association rules with 2 left hand attributes, e.g. $A, B \rightarrow C$, the $\chi^2$ test is performed by testing $B, C$ under all possible values of $A = \{a_1, a_2, ..., a_i\}$.
Appendix C

Bayesian networks

A Bayesian network (Heckerman et al., 1995; Heckerman, 1998; Pearl, 1998, 2000; Korb and Nicholson, 2003) consists of a set of nodes and a set of arcs (edges/links) which form a directed acyclic graph (DAG). Each node denotes one variable and an arc between two nodes represents the direct dependency between two variables. In the case of discrete variables, a child node\(^1\) of an arc is associated with a conditional probability table (CPT) representing its distribution under all values of the parent nodes. A skeleton of a Bayesian network is a graph of the network’s structure where the arcs are not directed. Since association rules deal only with discrete variables, the Bayesian networks discussed in the thesis consist only of nodes corresponding to variables with discrete values. An example of BN is shown in Figure 2.5.

A Bayesian network over a set of attributes \(H\) uniquely defines a joint probability distribution

\[
P_{H}^{BN} = \prod_{i=1}^{n} P_{A_i|\pi(A_i)},
\]

where \(A_i \in H\) is an attribute, and a vertex as well, and \(\pi(A_i)\) are the parents of \(A_i\).

---

\(^1\)A child node is at the end point of an arc.
Appendix D

Interactive Platform

The interactive platform is implemented in MATLAB™. We use the Bayes Net ToolBox for Matlab by Murphy (2001) as the inference engine of BNs. A screen capture of the application is displayed in Figure D.1. The procedure of the application is listed below:

1. The application loads the pre-mined unexpected rules and shows them in box (A) of Figure D.1. The assigned BK is also loaded into the memory.

2. The user clicks on a rule in box (A) for interpretation; another window of the local BN will pop-up to show the local BK (see Figure D.2 as an example).

3. Box (B) would show the found explanation (with highest score) of the selected rule. If there was no explanation found, it would show ‘no explanation found’.

4. The user has to give comments on the rule and the explanation in terms of surprisingness¹ and reasonableness in box (C).

5. The user could update the BK via selecting an attribute in box (D) and add one associated attribute in box (F) or delete an association in box (E).

6. Iterate steps 2 to 5 until all unexpected rules are interpreted or the user decides to stop.

¹The question in the screen shot is asking ‘unexpected’ rather than ‘surprising’ because the distinction of the two terms was developed after the implementation of the application; but the user’s comments would still be valid.
7. The user could press the [Find new unexpected rules] button to mine unexpected rules again with the updated BK.

Figure D.1: A screen capture of the implemented platform.
Figure D.2: An example of the local BN of a rule.
Appendix E

Sieve Diagrams

Riedwyl and Schüpbach (1994) proposed the ‘sieve diagram’ (also called as ‘parquet diagram’) to provide a direct, visual comparison of observed and expected frequencies under independence. A sieve diagram consists of rectangles that represent the marginal frequencies of two variables. Figure E.1 shows an example of sieve diagrams. The $x$-axis denotes the hair color and $y$-axis denotes the eye color. The height and width of each rectangle are proportional to the marginal frequencies of every value; the area of each rectangle is the expected frequency of its cell. The number of tiles (squares) in each rectangle (cell) represents the observed frequency in data; therefore, the denser the tiles, the higher observed frequency is. Let $f_o$ and $f_e$ be the observed and expected frequencies respectively; the color (red/blue) of rectangles is defined as following:

**Blue** (Positive deviation) $f_o > f_e$.

**Red** (Negative deviation) $f_o < f_e$.

The density of shading of the color (blue/red) is determined by the degree of deviation ($\frac{f_o}{f_e}$).
Figure E.1: Sieve diagram for hair-color, eye-color data. Observed frequencies are equal to the number squares in each cell, so departure from independence appears as variations in shading density; cited from (Friendly, 2001).
Appendix F

Strong Rule based Design for Knowledge Acquisition

In resolving the problem of ‘literal gaps’, we propose that presenting *strong* rules (assumption of exception rules, see Subsection 2.3.1) to a user is a practical start point for knowledge acquisition. Rules with high *support* and *confidence* would give the a user a strong impression about the data encoded under those attribute names; usually the rules also confirm the user’s knowledge. Dependent association rules (Subsection 5.3.2) could be further combined with the concept of strong rules to provide rules which are statistical dependent and strong associations. Therefore, instead of presenting the names of attributes and asking a user to assign associations based solely on attribute names, presenting strong dependent ARs at the beginning would be beneficial. We outline a possible design of knowledge acquisition which uses strong dependent ARs, which have high *support* and *confidence*, as the initial KB.

**Assumption 1.** A domain expert knows $\rho\%$, $\rho \approx 100$, of dependent ARs in data where the rules’ min-support $> \theta_1$ and min-confidence $> \theta_2$. In other words, $\rho\%$ of rules would be tagged as expected by the expert.

The assumption states that most dependent ARs ($\rho\%$) with high support and confidence are known by the expert. Based on the assumption, it is reasonable to use the known dependent ARs as the initial KB. Each rule $A \rightarrow B$ is treated as $A$ relates to $B$ and it adds a link in the association graph of KB. The hypothetical design of knowledge acquisition is plotted in Figure F.1. In the design, the association graph is built up from dependent ARs with high support and confidence; thus a user does not have to assign the association graph manually.
Figure F.1: A design of knowledge acquisition and unexpected rule mining based on strong dependent rules.
Appendix G

Preliminary Analysis of ANZDATA

The appendix comprises a collection of experiments of applying data mining to ANZDATA. Section G.1 presents the result of applying decision tree, emerging patterns (EP) and temporal ARs mining, and Section G.2 reports the result of association rules mining on ANZDATA.

G.1 Experiment I - Cardiovascular Death

The task of this experiment is to find the risk factors of cardiovascular death. At the beginning, two mining algorithms – REP Tree (Quinlan, 1992) and EP Miner (Ramamohanarao and Bailey, 2003; Ramamohanarao et al., 2005) – were used to have a quick analyze. All the different records of each patient are aggregated into one record regardless the temporal information. Variables are selected and expanded with little domain knowledge. As long as the variable is not totally irrelevant to cardiovascular death, it is kept in the transformed data otherwise it is removed. For example, the variable ‘total’ which means the total records of a patient is removed because it is only a redundant tag.

The result of REP decision tree is shown in Figure G.1. There is a strong predictor showing that when patients ever have cancer diagnosed, they are less likely to die of cardiovascular disease (CVD). When the weight is less than 38.15kg, they are less likely to die of CVD. On the other hand, if weight is more than 66.95kg, they are more likely to die of CVD. When the weight is between 38.15kg to 66.95kg, the likelihood of transplant becomes a influential predictor. This result should be analyzed more deeply before becoming more
G.1. EXPERIMENT I - CARDIOVASCULAR DEATH

Figure G.1: The result of REP Tree classifies cardiovascular death in ANZDATA.

meaningful. For example, ‘are patients children or underweight when they weight less than 38.15kg?’

Secondly, we applied EP Miner to the CardiovascularDeath class and two rules were mined:

- no CancerEverDiagnosed AND no RefluxNephropathy AND no AnalgesicNephropathy AND no LungDisease AND no Graft AND no LossFollowUp AND Height ≥ 1.512. 58% correct.

- no CancerEverDiagnosed AND no RefluxNephropathy AND no AnalgesicNephropathy AND no Transplant AND no LungDisease AND no LossFollowUp AND Height ≥ 1.512. 58% correct.

It shares the variable: CancerEverDiagnosed with the result of REP Tree.

Thirdly, in addressing the temporal aspect of data, we tried a temporal association miner. We implemented the algorithm of Winarko and Roddick (2005), applied it to the ANZDATA, and the result is shown below. The threshold of support and confidence is set to 30% and 70% respectively.

1. IF [no type1 diabetes] is finished by [2 dialysis sessions per week], THEN it is highly likely that [no type1 diabetes] equals [Alive or non-CV death] will also occur.
2. IF [Alive or Non-HF death] is finished by [Base weight=36-67], THEN it is highly likely that [Alive or Non-HF death] equals [Alive or non-CV death] will also occur.

3. IF [Alive or Non-HF death] is finished by [2 dialysis sessions per week], THEN it is highly likely that [Alive or Non-HF death] equals [Alive or non-CV death] will also occur.

An illustration of this result is shown in Figure G.2. However, this result is meaningless due to the overlapping meaning of variables. For example, [Alive or Non-HF death] is similar to [Alive or non-CV death]. This experiment indicates that either the temporal association miner by Winarko and Roddick (2005) is improper for ANZDATA or the data has to be preprocessed and transformed into proper format. Furthermore, there are many concepts that could not be represented by this algorithm such as the change of status in time. For example, the concept of ‘becoming obese in aging’ could not be represented.

G.1.1 Discussion

This experiment reveals the insufficiency of DM without temporal concepts, for example, there is an association in the DM result: ‘patients who do not receive renal transplant are highly likely to die of cardiovascular disease (CVD).’ However, this association overlooks the temporal concepts which are necessary for making more correct judgement such as diagnosed, before, after and so on. For example, this association would be more precise and interesting like this: ‘patients who are diagnosed with cardiovascular disease are highly likely to die of cardiovascular disease without receiving renal transplant.’ or ‘patients who are not diagnosed with cardiovascular disease are highly likely to die of
cardiovascular disease after receiving renal transplant.

Feedback from a nephrologist
The nephrologist, Dr. Paizis, commented on the results of REP decision tree and EP Miner and some of it is listed below.

- NoCancer → CardiovascularDeath
  According to the expert, people who have cancer usually die of cancer.

- NoTransplant → CardiovascularDeath
  For patients who do not receive transplant, they usually are very sick and some of them may have cardiovascular disease.

- NoRefluxNephropathy → CardiovascularDeath
  For patients who have RefluxNephropathy they are usually younger and have less cardiovascular disease.

- NoAnalgesicNephropathy → CardiovascularDeath
  This violates the normal knowledge because analgesic nephropathy is a risk factor of cardiovascular disease.

- (BodyMassIndex=18∼37) → CardiovascularDeath
  BMI=18∼37 is meaningless because most people fall within this range.

G.2 Experiment II - Association Rules

We implemented codes for association rule mining and applied it to the ANZ-DATA. There are 27,520 rules mined given minimum support = 0.1 and confidence = 0.6. Part of the rules, which have at least support = 0.6 and confidence = 0.8, are listed below. Unsurprisingly, the enormous number of rules is too large to be completely inspected by the expert. Therefore, it has little value from the user’s perspective.

Excerpt of Mined Association Rules

cancer=N → cerebrovascular=No, supp=0.79, conf=0.90
cancer=N → ESRD-Analgesicnephropathy=No, supp=0.79, conf=0.95
cancer=N → ESRD-PolycysticKD=No, supp=0.79, conf=0.94
cancer=N → ESRD-RefluxNephropathy=No, supp=0.79, conf=0.95
cancer=N → ESRD-Renovascular=No, supp=0.79, conf=0.88
APPENDIX G. PRELIMINARY ANALYSIS OF ANZDATA

cancer=N → hypertension=Yes, supp=0.79, conf=0.80
cancer=N → lungDisease=No, supp=0.79, conf=0.89
cancer=N → pvDisease=No, supp=0.79, conf=0.81
cerebrovascular=No → ESRD-Analgesicnephropathy=No, supp=0.90, conf=0.95
cerebrovascular=No → ESRD-PolycysticKD=No, supp=0.90, conf=0.93
cerebrovascular=No → ESRD-RefluxNephropathy=No, supp=0.90, conf=0.95
cerebrovascular=No → ESRD-Renovascular=No, supp=0.90, conf=0.90
cerebrovascular=No → lungDisease=No, supp=0.90, conf=0.90

diabetes=No → cerebrovascular=No, supp=0.69, conf=0.92
diabetes=No → ESRD-Analgesicnephropathy=No, supp=0.69, conf=0.93
diabetes=No → ESRD-Diabetes=No, supp=0.69, conf=1.00
diabetes=No → ESRD-PolycysticKD=No, supp=0.69, conf=0.91
diabetes=No → ESRD-RefluxNephropathy=No, supp=0.69, conf=0.93
diabetes=No → ESRD-Renovascular=No, supp=0.69, conf=0.86
diabetes=No → lungDisease=No, supp=0.69, conf=0.89
diabetes=No → pvDisease=No, supp=0.69, conf=0.88
diabetes=No → race=Caucasoid, supp=0.69, conf=0.85

ESRD-Analgesicnephropathy=No → cancer=N, supp=0.95, conf=0.80
ESRD-Analgesicnephropathy=No → cerebrovascular=No, supp=0.95, conf=0.91
ESRD-Analgesicnephropathy=No → ESRD-PolycysticKD=No, supp=0.95, conf=0.93
ESRD-Analgesicnephropathy=No → ESRD-RefluxNephropathy=No, supp=0.95, conf=0.95
ESRD-Analgesicnephropathy=No → ESRD-Renovascular=No, supp=0.95, conf=0.88
ESRD-Analgesicnephropathy=No → lungDisease=No, supp=0.95, conf=0.90
ESRD-Analgesicnephropathy=No → pvDisease=No, supp=0.95, conf=0.82
ESRD-Diabetes=No → cerebrovascular=No, supp=0.76, conf=0.91
ESRD-Diabetes=No → diabetes=No, supp=0.76, conf=0.90
ESRD-Diabetes=No → ESRD-Analgesicnephropathy=No, supp=0.76, conf=0.93
ESRD-Diabetes=No → ESRD-PolycysticKD=No, supp=0.76, conf=0.91
ESRD-Diabetes=No → ESRD-RefluxNephropathy=No, supp=0.76, conf=0.94
ESRD-Diabetes=No → ESRD-Renovascular=No, supp=0.76, conf=0.85
ESRD-Diabetes=No → lungDisease=No, supp=0.76, conf=0.89
ESRD-Diabetes=No → pvDisease=No, supp=0.76, conf=0.87
ESRD-Diabetes=No → race=Caucasoid, supp=0.76, conf=0.83
ESRD-GN=No → cerebrovascular=No, supp=0.69, conf=0.88
ESRD-GN=No → ESRD-Analgesicnephropathy=No, supp=0.69, conf=0.92
G.2. EXPERIMENT II - ASSOCIATION RULES

\[
\begin{align*}
\text{ESRD-GN} = \text{No} & \rightarrow \text{ESRD-PolycsticKD} = \text{No}, \ supp = 0.69, \ conf = 0.91 \\
\text{ESRD-GN} = \text{No} & \rightarrow \text{ESRD-RefluxNephropathy} = \text{No}, \ supp = 0.69, \ conf = 0.93 \\
\text{ESRD-GN} = \text{No} & \rightarrow \text{ESRD-Renovascular} = \text{No}, \ supp = 0.69, \ conf = 0.83 \\
\text{ESRD-GN} = \text{No} & \rightarrow \text{lungDisease} = \text{No}, \ supp = 0.69, \ conf = 0.88 \\
\text{ESRD-PolycsticKD} = \text{No} & \rightarrow \text{cerebrovascular} = \text{No}, \ supp = 0.93, \ conf = 0.90 \\
\text{ESRD-PolycsticKD} = \text{No} & \rightarrow \text{ESRD-Analgesicnephropathy} = \text{No}, \ supp = 0.93, \ conf = 0.94 \\
\text{ESRD-PolycsticKD} = \text{No} & \rightarrow \text{ESRD-RefluxNephropathy} = \text{No}, \ supp = 0.93, \ conf = 0.95 \\
\text{ESRD-PolycsticKD} = \text{No} & \rightarrow \text{ESRD-Renovascular} = \text{No}, \ supp = 0.93, \ conf = 0.88 \\
\text{ESRD-PolycsticKD} = \text{No} & \rightarrow \text{lungDisease} = \text{No}, \ supp = 0.93, \ conf = 0.89 \\
\text{ESRD-PolycsticKD} = \text{No} & \rightarrow \text{pvDisease} = \text{No}, \ supp = 0.93, \ conf = 0.80 \\
\text{ESRD-RefluxNephropathy} = \text{No} & \rightarrow \text{cerebrovascular} = \text{No}, \ supp = 0.95, \ conf = 0.90 \\
\text{ESRD-RefluxNephropathy} = \text{No} & \rightarrow \text{ESRD-Analgesicnephropathy} = \text{No}, \ supp = 0.95, \ conf = 0.94 \\
\text{ESRD-RefluxNephropathy} = \text{No} & \rightarrow \text{ESRD-PolycsticKD} = \text{No}, \ supp = 0.95, \ conf = 0.93 \\
\text{ESRD-RefluxNephropathy} = \text{No} & \rightarrow \text{ESRD-Renovascular} = \text{No}, \ supp = 0.95, \ conf = 0.88 \\
\text{ESRD-RefluxNephropathy} = \text{No} & \rightarrow \text{hypertension} = \text{Yes}, \ supp = 0.95, \ conf = 0.80 \\
\text{ESRD-RefluxNephropathy} = \text{No} & \rightarrow \text{lungDisease} = \text{No}, \ supp = 0.95, \ conf = 0.89 \\
\text{ESRD-Renovascular} = \text{No} & \rightarrow \text{cerebrovascular} = \text{No}, \ supp = 0.95, \ conf = 0.81 \\
\text{ESRD-Renovascular} = \text{No} & \rightarrow \text{pvDisease} = \text{No}, \ supp = 0.95, \ conf = 0.84 \\
\text{ESRD-Renovascular} = \text{No} & \rightarrow \text{hypertension} = \text{Yes}, \ supp = 0.88, \ conf = 0.90 \\
\text{ESRD-Renovascular} = \text{No} & \rightarrow \text{lungDisease} = \text{No}, \ supp = 0.88, \ conf = 0.90 \\
\text{ESRD-Renovascular} = \text{No} & \rightarrow \text{pvDisease} = \text{No}, \ supp = 0.88, \ conf = 0.84 \\
\text{frequency} = 3 & \rightarrow \text{cerebrovascular} = \text{No}, \ supp = 0.64, \ conf = 0.91 \\
\text{frequency} = 3 & \rightarrow \text{ESRD-Analgesicnephropathy} = \text{No}, \ supp = 0.64, \ conf = 0.95 \\
\text{frequency} = 3 & \rightarrow \text{ESRD-PolycsticKD} = \text{No}, \ supp = 0.64, \ conf = 0.93 \\
\text{frequency} = 3 & \rightarrow \text{ESRD-RefluxNephropathy} = \text{No}, \ supp = 0.64, \ conf = 0.95 \\
\text{frequency} = 3 & \rightarrow \text{ESRD-Renovascular} = \text{No}, \ supp = 0.64, \ conf = 0.89 \\
\text{frequency} = 3 & \rightarrow \text{hypertension} = \text{Yes}, \ supp = 0.64, \ conf = 0.81 \\
\text{frequency} = 3 & \rightarrow \text{lungDisease} = \text{No}, \ supp = 0.64, \ conf = 0.89 \\
\text{frequency} = 3 & \rightarrow \text{pvDisease} = \text{No}, \ supp = 0.64, \ conf = 0.83 \\
\text{hypertension} = \text{Yes} & \rightarrow \text{cancer} = \text{N}, \ supp = 0.80, \ conf = 0.80 \\
\text{hypertension} = \text{Yes} & \rightarrow \text{cerebrovascular} = \text{No}, \ supp = 0.80, \ conf = 0.90 \\
\text{hypertension} = \text{Yes} & \rightarrow \text{ESRD-Analgesicnephropathy} = \text{No}, \ supp = 0.80, \ conf = 0.95 \\
\text{hypertension} = \text{Yes} & \rightarrow \text{ESRD-PolycsticKD} = \text{No}, \ supp = 0.80, \ conf = 0.93 \\
\text{hypertension} = \text{Yes} & \rightarrow \text{ESRD-RefluxNephropathy} = \text{No}, \ supp = 0.80, \ conf = 0.96
\end{align*}
\]
hypertension=Yes $\rightarrow$ ESRD-Renovascular=No, supp=0.80, conf=0.87
hypertension=Yes $\rightarrow$ lungDisease=No, supp=0.80, conf=0.89
hypertension=Yes $\rightarrow$ pvDisease=No, supp=0.80, conf=0.81
lungDisease=No $\rightarrow$ cerebrovascular=No, supp=0.89, conf=0.91
lungDisease=No $\rightarrow$ ESRD-Analgesicnephropathy=No, supp=0.89, conf=0.95
lungDisease=No $\rightarrow$ ESRD-PolycsticKD=No, supp=0.89, conf=0.93
lungDisease=No $\rightarrow$ ESRD-RefluxNephropathy=No, supp=0.89, conf=0.95
lungDisease=No $\rightarrow$ ESRD-Renovascular=No, supp=0.89, conf=0.89
lungDisease=No $\rightarrow$ pvDisease=No, supp=0.89, conf=0.83
pvDisease=No $\rightarrow$ cerebrovascular=No, supp=0.81, conf=0.94
pvDisease=No $\rightarrow$ ESRD-Analgesicnephropathy=No, supp=0.81, conf=0.95
pvDisease=No $\rightarrow$ ESRD-Diabetes=No, supp=0.81, conf=0.81
pvDisease=No $\rightarrow$ ESRD-PolycsticKD=No, supp=0.81, conf=0.92
pvDisease=No $\rightarrow$ ESRD-RefluxNephropathy=No, supp=0.81, conf=0.94
pvDisease=No $\rightarrow$ ESRD-Renovascular=No, supp=0.81, conf=0.91
pvDisease=No $\rightarrow$ lungDisease=No, supp=0.81, conf=0.91
race=Caucasoid $\rightarrow$ cerebrovascular=No, supp=0.76, conf=0.90
race=Caucasoid $\rightarrow$ ESRD-Analgesicnephropathy=No, supp=0.76, conf=0.93
race=Caucasoid $\rightarrow$ ESRD-Diabetes=No, supp=0.76, conf=0.84
race=Caucasoid $\rightarrow$ ESRD-PolycsticKD=No, supp=0.76, conf=0.92
race=Caucasoid $\rightarrow$ ESRD-RefluxNephropathy=No, supp=0.76, conf=0.94
race=Caucasoid $\rightarrow$ ESRD-Renovascular=No, supp=0.76, conf=0.87
race=Caucasoid $\rightarrow$ lungDisease=No, supp=0.76, conf=0.89
race=Caucasoid $\rightarrow$ pvDisease=No, supp=0.76, conf=0.81
cancer=N, cerebrovascular=No $\rightarrow$ ESRD-Analgesicnephropathy=No, supp=0.72, conf=0.96
cancer=N, cerebrovascular=No $\rightarrow$ ESRD-PolycsticKD=No, supp=0.72, conf=0.94
cancer=N, cerebrovascular=No $\rightarrow$ ESRD-RefluxNephropathy=No, supp=0.72, conf=0.95
cancer=N, cerebrovascular=No $\rightarrow$ ESRD-Renovascular=No, supp=0.72, conf=0.90
cancer=N, cerebrovascular=No $\rightarrow$ hypertension=Yes, supp=0.72, conf=0.80
cancer=N, cerebrovascular=No $\rightarrow$ lungDisease=No, supp=0.72, conf=0.91
cancer=N, cerebrovascular=No $\rightarrow$ pvDisease=No, supp=0.72, conf=0.84
cancer=N, ESRD-Analgesicnephropathy=No $\rightarrow$ cerebrovascular=No, supp=0.76, conf=0.90
cancer=N, ESRD-Analgesicnephropathy=No $\rightarrow$ ESRD-PolycsticKD=No, supp=0.76, conf=0.94
cancer=N, ESRD-AnalgesicNephropathy=No → ESRD-RefluxNephropathy=No, supp=0.76, conf=0.95

cancer=N, ESRD-AnalgesicNephropathy=No → ESRD-Renovascular=No, supp=0.76, conf=0.88

cancer=N, ESRD-AnalgesicNephropathy=No → hypertension=Yes, supp=0.76, conf=0.80

cancer=N, ESRD-AnalgesicNephropathy=No → lungDisease=No, supp=0.76, conf=0.90

cancer=N, ESRD-AnalgesicNephropathy=No → pvDisease=No, supp=0.76, conf=0.81

cancer=N, ESRD-PolycysticKD=No → cerebrovascular=No, supp=0.75, conf=0.90

cancer=N, ESRD-PolycysticKD=No → ESRD-AnalgesicNephropathy=No, supp=0.75, conf=0.95

cancer=N, ESRD-PolycysticKD=No → ESRD-RefluxNephropathy=No, supp=0.75, conf=0.95

cancer=N, ESRD-PolycysticKD=No → ESRD-Renovascular=No, supp=0.75, conf=0.88

cancer=N, ESRD-PolycysticKD=No → hypertension=Yes, supp=0.75, conf=0.80

cancer=N, ESRD-PolycysticKD=No → lungDisease=No, supp=0.75, conf=0.89

cancer=N, ESRD-RefluxNephropathy=No → cerebrovascular=No, supp=0.76, conf=0.90

cancer=N, ESRD-RefluxNephropathy=No → ESRD-AnalgesicNephropathy=No, supp=0.76, conf=0.95

cancer=N, ESRD-RefluxNephropathy=No → ESRD-PolycysticKD=No, supp=0.76, conf=0.94

cancer=N, ESRD-RefluxNephropathy=No → ESRD-Renovascular=No, supp=0.76, conf=0.88

cancer=N, ESRD-RefluxNephropathy=No → hypertension=Yes, supp=0.76, conf=0.81

cancer=N, ESRD-RefluxNephropathy=No → lungDisease=No, supp=0.76, conf=0.89

cancer=N, ESRD-Renovascular=No → cerebrovascular=No, supp=0.70, conf=0.92

cancer=N, ESRD-Renovascular=No → ESRD-AnalgesicNephropathy=No, supp=0.70, conf=0.95

cancer=N, ESRD-Renovascular=No → ESRD-PolycysticKD=No, supp=0.70, conf=0.93

cancer=N, ESRD-Renovascular=No → ESRD-RefluxNephropathy=No, supp=0.70, conf=0.94

cancer=N, ESRD-Renovascular=No → lungDisease=No, supp=0.70, conf=0.90

cancer=N, ESRD-Renovascular=No → pvDisease=No, supp=0.70, conf=0.83

cancer=N, hypertension=Yes → cerebrovascular=No, supp=0.64, conf=0.90

cancer=N, hypertension=Yes → ESRD-AnalgesicNephropathy=No, supp=0.64, conf=0.95

cancer=N, hypertension=Yes → ESRD-PolycysticKD=No, supp=0.64, conf=0.94

cancer=N, hypertension=Yes → ESRD-RefluxNephropathy=No, supp=0.64, conf=0.96
cancer=N, hypertension=Yes → ESRD-Renovascular=No, supp=0.64, conf=0.87
cancer=N, hypertension=Yes → lungDisease=No, supp=0.64, conf=0.90
cancer=N, hypertension=Yes → pvDisease=No, supp=0.64, conf=0.80
cancer=N, lungDisease=No → cerebrovascular=No, supp=0.71, conf=0.91
cancer=N, lungDisease=No → ESRD-Analgesicnephropathy=No, supp=0.71, conf=0.96
cancer=N, lungDisease=No → ESRD-PolycsticKD=No, supp=0.71, conf=0.94
cancer=N, lungDisease=No → ESRD-RefluxNephropathy=No, supp=0.71, conf=0.95
cancer=N, lungDisease=No → ESRD-Renovascular=No, supp=0.71, conf=0.89
cancer=N, lungDisease=No → hypertension=Yes, supp=0.71, conf=0.81
cancer=N, lungDisease=No → pvDisease=No, supp=0.71, conf=0.83
cancer=N, pvDisease=No → cerebrovascular=No, supp=0.64, conf=0.94
cancer=N, pvDisease=No → ESRD-Analgesicnephropathy=No, supp=0.64, conf=0.95
cancer=N, pvDisease=No → ESRD-PolycsticKD=No, supp=0.64, conf=0.93
cancer=N, pvDisease=No → ESRD-RefluxNephropathy=No, supp=0.64, conf=0.94
cancer=N, pvDisease=No → ESRD-Renovascular=No, supp=0.64, conf=0.91
cancer=N, pvDisease=No → lungDisease=No, supp=0.64, conf=0.92
cerebrovascular=No, diabetes=No → ESRD-Analgesicnephropathy=No, supp=0.63, conf=0.93
cerebrovascular=No, diabetes=No → ESRD-Diabetes=No, supp=0.63, conf=1.00
cerebrovascular=No, diabetes=No → ESRD-PolycsticKD=No, supp=0.63, conf=0.91
cerebrovascular=No, diabetes=No → ESRD-RefluxNephropathy=No, supp=0.63, conf=0.93
cerebrovascular=No, diabetes=No → ESRD-Renovascular=No, supp=0.63, conf=0.87
cerebrovascular=No, diabetes=No → lungDisease=No, supp=0.63, conf=0.90
cerebrovascular=No, diabetes=No → pvDisease=No, supp=0.63, conf=0.91
cerebrovascular=No, diabetes=No → race=Caucasoid, supp=0.63, conf=0.84
cerebrovascular=No, ESRD-Analgesicnephropathy=No → ESRD-PolycsticKD=No, supp=0.86, conf=0.93
cerebrovascular=No, ESRD-Analgesicnephropathy=No → ESRD-RefluxNephropathy=No, supp=0.86, conf=0.95
cerebrovascular=No, ESRD-Analgesicnephropathy=No → ESRD-Renovascular=No, supp=0.86, conf=0.89
cerebrovascular=No, ESRD-Analgesicnephropathy=No → lungDisease=No, supp=0.86, conf=0.91
cerebrovascular=No, ESRD-Analgesicnephropathy=No → pvDisease=No, supp=0.86, conf=0.85
cerebrovascular=No, ESRD-Diabetes=No → diabetes=No, supp=0.70, conf=0.91

cerebrovascular=No, ESRD-Diabetes=No → ESRD-Analgesicnephropathy=No, supp=0.70, conf=0.93

cerebrovascular=No, ESRD-Diabetes=No → ESRD-PolycsticKD=No, supp=0.70, conf=0.91

cerebrovascular=No, ESRD-Diabetes=No → ESRD-RefluxNephropathy=No, supp=0.70, conf=0.93

cerebrovascular=No, ESRD-Diabetes=No → ESRD-Renovascular=No, supp=0.70, conf=0.87

cerebrovascular=No, ESRD-Diabetes=No → lungDisease=No, supp=0.70, conf=0.90

cerebrovascular=No, ESRD-Diabetes=No → pvDisease=No, supp=0.70, conf=0.90

cerebrovascular=No, ESRD-Diabetes=No → race=Caucasoid, supp=0.70, conf=0.83

cerebrovascular=No, ESRD-GN=No → ESRD-Analgesicnephropathy=No, supp=0.61, conf=0.92

cerebrovascular=No, ESRD-GN=No → ESRD-PolycsticKD=No, supp=0.61, conf=0.90

cerebrovascular=No, ESRD-GN=No → ESRD-RefluxNephropathy=No, supp=0.61, conf=0.92

cerebrovascular=No, ESRD-GN=No → ESRD-Renovascular=No, supp=0.61, conf=0.85

cerebrovascular=No, ESRD-GN=No → lungDisease=No, supp=0.61, conf=0.90

cerebrovascular=No, ESRD-GN=No → pvDisease=No, supp=0.61, conf=0.80

cerebrovascular=No, ESRD-PolycsticKD=No → ESRD-Analgesicnephropathy=No, supp=0.84, conf=0.94

cerebrovascular=No, ESRD-PolycsticKD=No → ESRD-RefluxNephropathy=No, supp=0.84, conf=0.94

cerebrovascular=No, ESRD-PolycsticKD=No → ESRD-Renovascular=No, supp=0.84, conf=0.89

cerebrovascular=No, ESRD-PolycsticKD=No → lungDisease=No, supp=0.84, conf=0.90

cerebrovascular=No, ESRD-PolycsticKD=No → pvDisease=No, supp=0.84, conf=0.84

cerebrovascular=No, ESRD-RefluxNephropathy=No → ESRD-Analgesicnephropathy=No, supp=0.86, conf=0.95

cerebrovascular=No, ESRD-RefluxNephropathy=No → ESRD-PolycsticKD=No, supp=0.86, conf=0.93

cerebrovascular=No, ESRD-RefluxNephropathy=No → ESRD-Renovascular=No, supp=0.86, conf=0.89

cerebrovascular=No, ESRD-RefluxNephropathy=No → lungDisease=No, supp=0.86, conf=0.90
APPENDIX G. PRELIMINARY ANALYSIS OF ANZDATA

cerebrovascular=No, ESRD-RefluxNephropathy=No → pvDisease=No, supp=0.86, conf=0.84
cerebrovascular=No, ESRD-Renovascular=No → ESRD-Analgesicnephropathy=No, supp=0.81, conf=0.94
cerebrovascular=No, ESRD-Renovascular=No → ESRD-PolycysticKD=No, supp=0.81, conf=0.93
cerebrovascular=No, ESRD-Renovascular=No → ESRD-RefluxNephropathy=No, supp=0.81, conf=0.94
cerebrovascular=No, ESRD-Renovascular=No → lungDisease=No, supp=0.81, conf=0.91
cerebrovascular=No, ESRD-Renovascular=No → pvDisease=No, supp=0.81, conf=0.86
cerebrovascular=No, hypertension=Yes → cancer=N, supp=0.72, conf=0.80
cerebrovascular=No, hypertension=Yes → ESRD-Analgesicnephropathy=No, supp=0.72, conf=0.95
cerebrovascular=No, hypertension=Yes → ESRD-PolycysticKD=No, supp=0.72, conf=0.93
cerebrovascular=No, hypertension=Yes → ESRD-RefluxNephropathy=No, supp=0.72, conf=0.95
cerebrovascular=No, hypertension=Yes → ESRD-Renovascular=No, supp=0.72, conf=0.89
cerebrovascular=No, hypertension=Yes → lungDisease=No, supp=0.72, conf=0.90
cerebrovascular=No, hypertension=Yes → pvDisease=No, supp=0.72, conf=0.84
cerebrovascular=No, lungDisease=No → ESRD-Analgesicnephropathy=No, supp=0.81, conf=0.95
cerebrovascular=No, lungDisease=No → ESRD-PolycysticKD=No, supp=0.81, conf=0.93
cerebrovascular=No, lungDisease=No → ESRD-RefluxNephropathy=No, supp=0.81, conf=0.94
cerebrovascular=No, lungDisease=No → ESRD-Renovascular=No, supp=0.81, conf=0.90
cerebrovascular=No, lungDisease=No → pvDisease=No, supp=0.81, conf=0.86
cerebrovascular=No, pvDisease=No → ESRD-Analgesicnephropathy=No, supp=0.76, conf=0.95
cerebrovascular=No, pvDisease=No → ESRD-Diabetes=No, supp=0.76, conf=0.82
cerebrovascular=No, pvDisease=No → ESRD-PolycysticKD=No, supp=0.76, conf=0.92
cerebrovascular=No, pvDisease=No → ESRD-RefluxNephropathy=No, supp=0.76, conf=0.94
cerebrovascular=No, pvDisease=No → ESRD-Renovascular=No, supp=0.76, conf=0.92
cerebrovascular=No, pvDisease=No → lungDisease=No, supp=0.76, conf=0.92
cerebrovascular=No, race=Caucasoid → ESRD-Analgesicnephropathy=No, supp=0.68, conf=0.93
cerebrovascular=No, race=Caucasoid → ESRD-Diabetes=No, supp=0.68, conf=0.85
cerebrovascular=No, race=Caucasoid → ESRD-PolycysticKD=No, supp=0.68, conf=0.92
cerebrovascular=No, race=Caucasoid → ESRD-RefluxNephropathy=No, supp=0.68, conf=0.94
cerebrovascular=No, race=Caucasoid → ESRD-Renovascular=No, supp=0.68, conf=0.89
cerebrovascular=No, race=Caucasoid → lungDisease=No, supp=0.68, conf=0.90
cerebrovascular=No, race=Caucasoid → pvDisease=No, supp=0.68, conf=0.85
diabetes=No, ESRD-Analgesicnephropathy=No → cerebrovascular=No, supp=0.64, conf=0.92
diabetes=No, ESRD-Analgesicnephropathy=No → ESRD-Diabetes=No, supp=0.64, conf=1.00
diabetes=No, ESRD-Analgesicnephropathy=No → ESRD-PolycysticKD=No, supp=0.64, conf=0.90
diabetes=No, ESRD-Analgesicnephropathy=No → ESRD-RefluxNephropathy=No, supp=0.64, conf=0.93
diabetes=No, ESRD-Analgesicnephropathy=No → ESRD-Renovascular=No, supp=0.64, conf=0.84
diabetes=No, ESRD-Analgesicnephropathy=No → lungDisease=No, supp=0.64, conf=0.90
diabetes=No, ESRD-Analgesicnephropathy=No → pvDisease=No, supp=0.64, conf=0.89
diabetes=No, ESRD-Diabetes=No → race=Caucasoid, supp=0.64, conf=0.84
diabetes=No, ESRD-Diabetes=No → cerebrovascular=No, supp=0.69, conf=0.92
diabetes=No, ESRD-Diabetes=No → ESRD-Analgesicnephropathy=No, supp=0.69, conf=0.93
diabetes=No, ESRD-Diabetes=No → ESRD-PolycysticKD=No, supp=0.69, conf=0.91
diabetes=No, ESRD-Diabetes=No → ESRD-RefluxNephropathy=No, supp=0.69, conf=0.93
diabetes=No, ESRD-Diabetes=No → ESRD-Renovascular=No, supp=0.69, conf=0.86
diabetes=No, ESRD-Diabetes=No → lungDisease=No, supp=0.69, conf=0.89
diabetes=No, ESRD-Diabetes=No → pvDisease=No, supp=0.69, conf=0.88
diabetes=No, ESRD-Diabetes=No → race=Caucasoid, supp=0.69, conf=0.85
diabetes=No, ESRD-PolycysticKD=No → cerebrovascular=No, supp=0.63, conf=0.92
diabetes=No, ESRD-PolycysticKD=No → ESRD-Analgesicnephropathy=No, supp=0.63, conf=0.92
diabetes=No, ESRD-PolycysticKD=No → ESRD-Diabetes=No, supp=0.63, conf=1.00
diabetes=No, ESRD-PolycysticKD=No → ESRD-RefluxNephropathy=No, supp=0.63, conf=0.93
diabetes=No, ESRD-PolycsticKD=No → ESRD-Renovascular=No, supp=0.63, conf=0.84

diabetes=No, ESRD-PolycsticKD=No → lungDisease=No, supp=0.63, conf=0.89

diabetes=No, ESRD-PolycsticKD=No → pvDisease=No, supp=0.63, conf=0.88

diabetes=No, ESRD-PolycsticKD=No → race=Caucasoid, supp=0.63, conf=0.84

diabetes=No, ESRD-RefluxNephropathy=No → cerebrovascular=No, supp=0.64, conf=0.91

diabetes=No, ESRD-RefluxNephropathy=No → ESRD-Analgesicnephropathy=No, supp=0.64, conf=0.92

diabetes=No, ESRD-RefluxNephropathy=No → ESRD-Diabetes=No, supp=0.64, conf=1.00

diabetes=No, ESRD-RefluxNephropathy=No → ESRD-PolycsticKD=No, supp=0.64, conf=0.90

diabetes=No, ESRD-RefluxNephropathy=No → ESRD-Renovascular=No, supp=0.64, conf=0.85

diabetes=No, ESRD-RefluxNephropathy=No → lungDisease=No, supp=0.64, conf=0.89

diabetes=No, ESRD-RefluxNephropathy=No → pvDisease=No, supp=0.64, conf=0.88

diabetes=No, ESRD-RefluxNephropathy=No → race=Caucasoid, supp=0.64, conf=0.84

diabetes=No, lungDisease=No → cerebrovascular=No, supp=0.61, conf=0.93

diabetes=No, lungDisease=No → ESRD-Analgesicnephropathy=No, supp=0.61, conf=0.94

diabetes=No, lungDisease=No → ESRD-Diabetes=No, supp=0.61, conf=1.00

diabetes=No, lungDisease=No → ESRD-PolycsticKD=No, supp=0.61, conf=0.90

diabetes=No, lungDisease=No → ESRD-RefluxNephropathy=No, supp=0.61, conf=0.93

diabetes=No, lungDisease=No → ESRD-Renovascular=No, supp=0.61, conf=0.86

diabetes=No, lungDisease=No → pvDisease=No, supp=0.61, conf=0.90

diabetes=No, lungDisease=No → race=Caucasoid, supp=0.61, conf=0.84

diabetes=No, pvDisease=No → cerebrovascular=No, supp=0.61, conf=0.95

diabetes=No, pvDisease=No → ESRD-Analgesicnephropathy=No, supp=0.61, conf=0.94

diabetes=No, pvDisease=No → ESRD-Diabetes=No, supp=0.61, conf=1.00

diabetes=No, pvDisease=No → ESRD-PolycsticKD=No, supp=0.61, conf=0.90

diabetes=No, pvDisease=No → ESRD-RefluxNephropathy=No, supp=0.61, conf=0.93

diabetes=No, pvDisease=No → ESRD-Renovascular=No, supp=0.61, conf=0.89

EsrD-Analgesicnephropathy=No, ESRD-Diabetes=No → cerebrovascular=No, supp=0.71, conf=0.92

EsrD-Analgesicnephropathy=No, ESRD-Diabetes=No → diabetes=No, supp=0.71,
conf=0.90
ESRD-Analgesicnephropathy=No, ESRD-Diabetes=No → ESRD-PolycsticKD=No, supp=0.71, conf=0.91
ESRD-Analgesicnephropathy=No, ESRD-Diabetes=No → ESRD-RefluxNephropathy=No, supp=0.71, conf=0.93
ESRD-Analgesicnephropathy=No, ESRD-Diabetes=No → ESRD-Renovascular=No, supp=0.71, conf=0.84
ESRD-Analgesicnephropathy=No, ESRD-Diabetes=No → lungDisease=No, supp=0.71, conf=0.89
ESRD-Analgesicnephropathy=No, ESRD-Diabetes=No → pvDisease=No, supp=0.71, conf=0.87
ESRD-Analgesicnephropathy=No, ESRD-Diabetes=No → race=Caucasoid, supp=0.71, conf=0.82
ESRD-Analgesicnephropathy=No, ESRD-GN=No → cancer=N, supp=0.63, conf=0.80
ESRD-Analgesicnephropathy=No, ESRD-GN=No → cerebrovascular=No, supp=0.63, conf=0.89
ESRD-Analgesicnephropathy=No, ESRD-GN=No → ESRD-PolycsticKD=No, supp=0.63, conf=0.90
ESRD-Analgesicnephropathy=No, ESRD-GN=No → ESRD-RefluxNephropathy=No, supp=0.63, conf=0.92
ESRD-Analgesicnephropathy=No, ESRD-GN=No → ESRD-Renovascular=No, supp=0.63, conf=0.82
ESRD-Analgesicnephropathy=No, ESRD-GN=No → lungDisease=No, supp=0.63, conf=0.89
ESRD-Analgesicnephropathy=No, ESRD-PolycsticKD=No → cancer=N, supp=0.88, conf=0.81
ESRD-Analgesicnephropathy=No, ESRD-PolycsticKD=No → cerebrovascular=No, supp=0.88, conf=0.90
ESRD-Analgesicnephropathy=No, ESRD-PolycsticKD=No → ESRD-RefluxNephropathy=No, supp=0.88, conf=0.95
ESRD-Analgesicnephropathy=No, ESRD-PolycsticKD=No → ESRD-Renovascular=No, supp=0.88, conf=0.87
ESRD-Analgesicnephropathy=No, ESRD-PolycsticKD=No → lungDisease=No, supp=0.88, conf=0.89
ESRD-Analgesicnephropathy=No, ESRD-PolycsticKD=No → pvDisease=No, supp=0.88, conf=0.81
ESRD-Analgesicnephropathy=No, ESRD-RefluxNephropathy=No → cerebrovascular=No, supp=0.90, conf=0.90
ESRD-Analgesicnephropathy=No, ESRD-RefluxNephropathy=No → ESRD-PolycysticKD=No, supp=0.90, conf=0.93
ESRD-Analgesicnephropathy=No, ESRD-RefluxNephropathy=No → ESRD-Renovascular=No, supp=0.90, conf=0.87
ESRD-Analgesicnephropathy=No, ESRD-RefluxNephropathy=No → hypertension=Yes, supp=0.90, conf=0.80
ESRD-Analgesicnephropathy=No, ESRD-RefluxNephropathy=No → lungDisease=No, supp=0.90, conf=0.89
ESRD-Analgesicnephropathy=No, ESRD-RefluxNephropathy=No → pvDisease=No, supp=0.90, conf=0.81
ESRD-Analgesicnephropathy=No, frequency=3 → cerebrovascular=No, supp=0.60, conf=0.91
ESRD-Analgesicnephropathy=No, frequency=3 → ESRD-PolycysticKD=No, supp=0.60, conf=0.92
ESRD-Analgesicnephropathy=No, frequency=3 → ESRD-RefluxNephropathy=No, supp=0.60, conf=0.95
ESRD-Analgesicnephropathy=No, frequency=3 → ESRD-Renovascular=No, supp=0.60, conf=0.88
ESRD-Analgesicnephropathy=No, frequency=3 → hypertension=Yes, supp=0.60, conf=0.81
ESRD-Analgesicnephropathy=No, frequency=3 → lungDisease=No, supp=0.60, conf=0.90
ESRD-Analgesicnephropathy=No, frequency=3 → pvDisease=No, supp=0.60, conf=0.83
G.2. EXPERIMENT II - ASSOCIATION RULES

ESRD-Analgesicnephropathy=No, hypertension=Yes → cancer=N, supp=0.75, conf=0.81
ESRD-Analgesicnephropathy=No, hypertension=Yes → cerebrovascular=No, supp=0.75, conf=0.90
ESRD-Analgesicnephropathy=No, hypertension=Yes → ESRD-PolycysticKD=No, supp=0.75, conf=0.93
ESRD-Analgesicnephropathy=No, hypertension=Yes → ESRD-RefluxNephropathy=No, supp=0.75, conf=0.95
ESRD-Analgesicnephropathy=No, hypertension=Yes → ESRD-Renovascular=No, supp=0.75, conf=0.87
ESRD-Analgesicnephropathy=No, hypertension=Yes → lungDisease=No, supp=0.75, conf=0.90
ESRD-Analgesicnephropathy=No, hypertension=Yes → pvDisease=No, supp=0.75, conf=0.81
ESRD-Analgesicnephropathy=No, lungDisease=No → cancer=N, supp=0.85, conf=0.80
ESRD-Analgesicnephropathy=No, lungDisease=No → cerebrovascular=No, supp=0.85, conf=0.92
ESRD-Analgesicnephropathy=No, lungDisease=No → ESRD-PolycysticKD=No, supp=0.85, conf=0.93
ESRD-Analgesicnephropathy=No, lungDisease=No → ESRD-RefluxNephropathy=No, supp=0.85, conf=0.95
ESRD-Analgesicnephropathy=No, lungDisease=No → ESRD-Renovascular=No, supp=0.85, conf=0.89
ESRD-Analgesicnephropathy=No, lungDisease=No → pvDisease=No, supp=0.85, conf=0.84
ESRD-Analgesicnephropathy=No, pvDisease=No → cerebrovascular=No, supp=0.77, conf=0.94
ESRD-Analgesicnephropathy=No, pvDisease=No → ESRD-Diabetes=No, supp=0.77, conf=0.80
ESRD-Analgesicnephropathy=No, pvDisease=No → ESRD-PolycysticKD=No, supp=0.77, conf=0.92
ESRD-Analgesicnephropathy=No, pvDisease=No → ESRD-RefluxNephropathy=No, supp=0.77, conf=0.94
ESRD-Analgesicnephropathy=No, pvDisease=No → ESRD-Renovascular=No, supp=0.77, conf=0.91
ESRD-Analgesicnephropathy=No, pvDisease=No → lungDisease=No, supp=0.77, conf=0.92
ESRD-Analgesicnephropathy=No, race=Caucasoid → cerebrovascular=No, supp=0.71, conf=0.90
ESRD-Analgesicnephropathy=No, race=Caucasoid → ESRD-Diabetes=No, supp=0.71, conf=0.83
ESRD-Analgesicnephropathy=No, race=Caucasoid → ESRD-PolycsticKD=No, supp=0.71, conf=0.91
ESRD-Analgesicnephropathy=No, race=Caucasoid → ESRD-RefluxNephropathy=No, supp=0.71, conf=0.94
ESRD-Analgesicnephropathy=No, race=Caucasoid → ESRD-Renovascular=No, supp=0.71, conf=0.86
ESRD-Analgesicnephropathy=No, race=Caucasoid → lungDisease=No, supp=0.71, conf=0.89
ESRD-Analgesicnephropathy=No, race=Caucasoid → pvDisease=No, supp=0.71, conf=0.81
ESRD-Diabetes=No, ESRD-PolycsticKD=No → cerebrovascular=No, supp=0.70, conf=0.91
ESRD-Diabetes=No, ESRD-PolycsticKD=No → diabetes=No, supp=0.70, conf=0.90
ESRD-Diabetes=No, ESRD-PolycsticKD=No → ESRD-Analgesicnephropathy=No, supp=0.70, conf=0.92
ESRD-Diabetes=No, ESRD-PolycsticKD=No → ESRD-RefluxNephropathy=No, supp=0.70, conf=0.93
ESRD-Diabetes=No, ESRD-PolycsticKD=No → ESRD-Renovascular=No, supp=0.70, conf=0.83
ESRD-Diabetes=No, ESRD-PolycsticKD=No → lungDisease=No, supp=0.70, conf=0.88
ESRD-Diabetes=No, ESRD-PolycsticKD=No → pvDisease=No, supp=0.70, conf=0.86
ESRD-Diabetes=No, ESRD-PolycsticKD=No → race=Caucasoid, supp=0.70, conf=0.82
ESRD-Diabetes=No, ESRD-RefluxNephropathy=No → cerebrovascular=No, supp=0.72, conf=0.91
ESRD-Diabetes=No, ESRD-RefluxNephropathy=No → diabetes=No, supp=0.72, conf=0.90
ESRD-Diabetes=No, ESRD-RefluxNephropathy=No → ESRD-Analgesicnephropathy=No, supp=0.72, conf=0.92
ESRD-Diabetes=No, ESRD-RefluxNephropathy=No → ESRD-PolycsticKD=No, supp=0.72, conf=0.91
ESRD-Diabetes=No, ESRD-RefluxNephropathy=No → ESRD-Renovascular=No, supp=0.72, conf=0.84
ESRD-Diabetes=No, ESRD-RefluxNephropathy=No → lungDisease=No, supp=0.72, conf=0.88
ESRD-Diabetes=No, ESRD-RefluxNephropathy=No → pvDisease=No, supp=0.72, conf=0.86
ESRD-Diabetes=No, ESRD-RefluxNephropathy=No → race=Caucasoid, supp=0.72, conf=0.83
ESRD-Diabetes=No, ESRD-Renovascular=No → cerebrovascular=No, supp=0.65, conf=0.93
ESRD-Diabetes=No, ESRD-Renovascular=No → diabetes=No, supp=0.65, conf=0.91
ESRD-Diabetes=No, ESRD-Renovascular=No → ESRD-Analgesicnephropathy=No, supp=0.65, conf=0.92
ESRD-Diabetes=No, ESRD-Renovascular=No → ESRD-PolycsticKD=No, supp=0.65, conf=0.90
ESRD-Diabetes=No, ESRD-Renovascular=No → ESRD-RefluxNephropathy=No, supp=0.65, conf=0.93
ESRD-Diabetes=No, ESRD-Renovascular=No → lungDisease=No, supp=0.65, conf=0.90
ESRD-Diabetes=No, ESRD-Renovascular=No → pvDisease=No, supp=0.65, conf=0.91
ESRD-Diabetes=No, ESRD-Renovascular=No → race=Caucasoid, supp=0.65, conf=0.83
ESRD-Diabetes=No, lungDisease=No → cerebrovascular=No, supp=0.68, conf=0.92
ESRD-Diabetes=No, lungDisease=No → diabetes=No, supp=0.68, conf=0.91
ESRD-Diabetes=No, lungDisease=No → ESRD-Analgesicnephropathy=No, supp=0.68, conf=0.94
ESRD-Diabetes=No, lungDisease=No → ESRD-PolycsticKD=No, supp=0.68, conf=0.91
ESRD-Diabetes=No, lungDisease=No → ESRD-RefluxNephropathy=No, supp=0.68, conf=0.93
ESRD-Diabetes=No, lungDisease=No → ESRD-Renovascular=No, supp=0.68, conf=0.86
ESRD-Diabetes=No, lungDisease=No → pvDisease=No, supp=0.68, conf=0.89
ESRD-Diabetes=No, pvDisease=No → race=Caucasoid, supp=0.68, conf=0.83
ESRD-Diabetes=No, pvDisease=No → cerebrovascular=No, supp=0.66, conf=0.94
ESRD-Diabetes=No, pvDisease=No → diabetes=No, supp=0.66, conf=0.92
ESRD-Diabetes=No, pvDisease=No → ESRD-Analgesicnephropathy=No, supp=0.66, conf=0.94
ESRD-Diabetes=No, pvDisease=No → ESRD-PolycsticKD=No, supp=0.66, conf=0.91
ESRD-Diabetes=No, pvDisease=No → ESRD-RefluxNephropathy=No, supp=0.66, conf=0.93
ESRD-Diabetes=No, pvDisease=No → ESRD-Renovascular=No, supp=0.66, conf=0.89
APPENDIX G. PRELIMINARY ANALYSIS OF ANZDATA

ESRD-Diabetes=No, pvDisease=No → lungDisease=No, supp=0.66, conf=0.91
ESRD-Diabetes=No, pvDisease=No → race=Caucasoid, supp=0.66, conf=0.82
ESRD-Diabetes=No, race=Caucasoid → cerebrovascular=No, supp=0.64, conf=0.91
ESRD-Diabetes=No, race=Caucasoid → diabetes=No, supp=0.64, conf=0.92
ESRD-Diabetes=No, race=Caucasoid → ESRD-Analgesicnephropathy=No, supp=0.64, conf=0.92
ESRD-Diabetes=No, race=Caucasoid → ESRD-PolycysticKD=No, supp=0.64, conf=0.90
ESRD-Diabetes=No, race=Caucasoid → ESRD-RefluxNephropathy=No, supp=0.64, conf=0.93
ESRD-Diabetes=No, race=Caucasoid → ESRD-Renovascular=No, supp=0.64, conf=0.84
ESRD-Diabetes=No, race=Caucasoid → lungDisease=No, supp=0.64, conf=0.88
ESRD-Diabetes=No, race=Caucasoid → pvDisease=No, supp=0.64, conf=0.86
ESRD-GN=No, ESRD-PolycysticKD=No → cancer=N, supp=0.62, conf=0.80
ESRD-GN=No, ESRD-PolycysticKD=No → cerebrovascular=No, supp=0.62, conf=0.88
ESRD-GN=No, ESRD-PolycysticKD=No → ESRD-Analgesicnephropathy=No, supp=0.62, conf=0.91
ESRD-GN=No, ESRD-PolycysticKD=No → ESRD-RefluxNephropathy=No, supp=0.62, conf=0.92
ESRD-GN=No, ESRD-PolycysticKD=No → ESRD-Renovascular=No, supp=0.62, conf=0.81
ESRD-GN=No, ESRD-PolycysticKD=No → lungDisease=No, supp=0.62, conf=0.88
ESRD-GN=No, ESRD-RefluxNephropathy=No → cerebrovascular=No, supp=0.64, conf=0.88
ESRD-GN=No, ESRD-RefluxNephropathy=No → ESRD-Analgesicnephropathy=No, supp=0.64, conf=0.92
ESRD-GN=No, ESRD-RefluxNephropathy=No → ESRD-PolycysticKD=No, supp=0.64, conf=0.90
ESRD-GN=No, ESRD-RefluxNephropathy=No → ESRD-Renovascular=No, supp=0.64, conf=0.82
ESRD-GN=No, ESRD-RefluxNephropathy=No → lungDisease=No, supp=0.64, conf=0.88
ESRD-GN=No, lungDisease=No → cerebrovascular=No, supp=0.61, conf=0.90
ESRD-GN=No, lungDisease=No → ESRD-Analgesicnephropathy=No, supp=0.61, conf=0.93
ESRD-GN=No, lungDisease=No → ESRD-PolycysticKD=No, supp=0.61, conf=0.90
ESRD-GN=No, lungDisease=No → ESRD-RefluxNephropathy=No, supp=0.61, conf=0.92
ESRD-GN=No, lungDisease=No → ESRD-Renovascular=No, supp=0.61, conf=0.84
G.2. EXPERIMENT II - ASSOCIATION RULES

ESRD-PolycysticKD=No, ESRD-RefluxNephropathy=No → cerebrovascular=No, supp=0.89, conf=0.90
ESRD-PolycysticKD=No, ESRD-RefluxNephropathy=No → ESRD-Analgesicnephropathy=No, supp=0.89, conf=0.94
ESRD-PolycysticKD=No, ESRD-RefluxNephropathy=No → ESRD-Renovascular=No, supp=0.89, conf=0.87
ESRD-PolycysticKD=No, ESRD-RefluxNephropathy=No → lungDisease=No, supp=0.88, conf=0.88
ESRD-PolycysticKD=No, ESRD-Renovascular=No → cancer=N, supp=0.82, conf=0.80
ESRD-PolycysticKD=No, ESRD-Renovascular=No → cerebrovascular=No, supp=0.82, conf=0.92
ESRD-PolycysticKD=No, ESRD-Renovascular=No → ESRD-Analgesicnephropathy=No, supp=0.82, conf=0.93
ESRD-PolycysticKD=No, ESRD-Renovascular=No → ESRD-RefluxNephropathy=No, supp=0.82, conf=0.94
ESRD-PolycysticKD=No, ESRD-Renovascular=No → lungDisease=No, supp=0.82, conf=0.90
ESRD-PolycysticKD=No, ESRD-Renovascular=No → pvDisease=No, supp=0.82, conf=0.83
ESRD-PolycysticKD=No, hypertension=Yes → cancer=N, supp=0.74, conf=0.81
ESRD-PolycysticKD=No, hypertension=Yes → cerebrovascular=No, supp=0.74, conf=0.90
ESRD-PolycysticKD=No, hypertension=Yes → ESRD-Analgesicnephropathy=No, supp=0.74, conf=0.94
ESRD-PolycysticKD=No, hypertension=Yes → ESRD-RefluxNephropathy=No, supp=0.74, conf=0.95
ESRD-PolycysticKD=No, hypertension=Yes → ESRD-Renovascular=No, supp=0.74, conf=0.86
ESRD-PolycysticKD=No, lungDisease=No → cancer=N, supp=0.83, conf=0.80
ESRD-PolycysticKD=No, lungDisease=No → cerebrovascular=No, supp=0.83, conf=0.91
ESRD-PolycysticKD=No, lungDisease=No → ESRD-Analgesicnephropathy=No, supp=0.83, conf=0.95
ESRD-PolycysticKD=No, lungDisease=No → ESRD-RefluxNephropathy=No, supp=0.83, conf=0.94
ESRD-PolycysticKD=No, lungDisease=No → ESRD-Renovascular=No, supp=0.83, conf=0.88
ESRD-PolycsticKD=No, lungDisease=No → pvDisease=No, supp=0.83, conf=0.83
ESRD-PolycsticKD=No, pvDisease=No → cerebrovascular=No, supp=0.75, conf=0.94
ESRD-PolycsticKD=No, pvDisease=No → ESRD-Analgesicnephropathy=No, supp=0.75, conf=0.94
ESRD-PolycsticKD=No, pvDisease=No → ESRD-RefluxNephropathy=No, supp=0.75, conf=0.94
ESRD-PolycsticKD=No, pvDisease=No → ESRD-Renovascular=No, supp=0.75, conf=0.90
ESRD-PolycsticKD=No, pvDisease=No → lungDisease=No, supp=0.75, conf=0.91
ESRD-PolycsticKD=No, race=Caucasoid → cerebrovascular=No, supp=0.70, conf=0.89
ESRD-PolycsticKD=No, race=Caucasoid → ESRD-Analgesicnephropathy=No, supp=0.70, conf=0.92
ESRD-PolycsticKD=No, race=Caucasoid → ESRD-Diabetes=No, supp=0.70, conf=0.82
ESRD-PolycsticKD=No, race=Caucasoid → ESRD-RefluxNephropathy=No, supp=0.70, conf=0.94
ESRD-PolycsticKD=No, race=Caucasoid → ESRD-Renovascular=No, supp=0.70, conf=0.86
ESRD-PolycsticKD=No, race=Caucasoid → lungDisease=No, supp=0.70, conf=0.88
ESRD-RefluxNephropathy=No, ESRD-Renovascular=No → cerebrovascular=No, supp=0.84, conf=0.91
ESRD-RefluxNephropathy=No, ESRD-Renovascular=No → ESRD-Analgesicnephropathy=No, supp=0.84, conf=0.94
ESRD-RefluxNephropathy=No, ESRD-Renovascular=No → ESRD-PolycsticKD=No, supp=0.84, conf=0.92
ESRD-RefluxNephropathy=No, ESRD-Renovascular=No → lungDisease=No, supp=0.84, conf=0.90
ESRD-RefluxNephropathy=No, ESRD-Renovascular=No → pvDisease=No, supp=0.84, conf=0.83
ESRD-RefluxNephropathy=No, frequency=3 → cerebrovascular=No, supp=0.61, conf=0.91
ESRD-RefluxNephropathy=No, frequency=3 → ESRD-Analgesicnephropathy=No, supp=0.61, conf=0.94
ESRD-RefluxNephropathy=No, frequency=3 → ESRD-PolycsticKD=No, supp=0.61, conf=0.92
ESRD-RefluxNephropathy=No, frequency=3 → ESRD-Renovascular=No, supp=0.61, conf=0.88
G.2. EXPERIMENT II - ASSOCIATION RULES

ESRD-RefluxNephropathy=No, frequency=3 → hypertension=Yes, supp=0.61, conf=0.81
ESRD-RefluxNephropathy=No, frequency=3 → lungDisease=No, supp=0.61, conf=0.89
ESRD-RefluxNephropathy=No, frequency=3 → pvDisease=No, supp=0.61, conf=0.82
ESRD-RefluxNephropathy=No, hypertension=Yes → cancer=N, supp=0.76, conf=0.80
ESRD-RefluxNephropathy=No, hypertension=Yes → cerebrovascular=No, supp=0.76, conf=0.89
ESRD-RefluxNephropathy=No, hypertension=Yes → ESRD-Analgesicnephropathy=No, supp=0.76, conf=0.94
ESRD-RefluxNephropathy=No, hypertension=Yes → ESRD-PolycsticKD=No, supp=0.76, conf=0.93
ESRD-RefluxNephropathy=No, hypertension=Yes → ESRD-Renovascular=No, supp=0.76, conf=0.87
ESRD-RefluxNephropathy=No, hypertension=Yes → lungDisease=No, supp=0.76, conf=0.89
ESRD-RefluxNephropathy=No, lungDisease=No → cerebrovascular=No, supp=0.84, conf=0.91
ESRD-RefluxNephropathy=No, lungDisease=No → ESRD-Analgesicnephropathy=No, supp=0.84, conf=0.95
ESRD-RefluxNephropathy=No, lungDisease=No → ESRD-PolycsticKD=No, supp=0.84, conf=0.93
ESRD-RefluxNephropathy=No, lungDisease=No → ESRD-Renovascular=No, supp=0.84, conf=0.89
ESRD-RefluxNephropathy=No, lungDisease=No → hypertension=Yes, supp=0.84, conf=0.80
ESRD-RefluxNephropathy=No, lungDisease=No → pvDisease=No, supp=0.84, conf=0.83
ESRD-RefluxNephropathy=No, pvDisease=No → cerebrovascular=No, supp=0.77, conf=0.94
ESRD-RefluxNephropathy=No, pvDisease=No → ESRD-Analgesicnephropathy=No, supp=0.77, conf=0.95
ESRD-RefluxNephropathy=No, pvDisease=No → ESRD-Diabetes=No, supp=0.77, conf=0.80
ESRD-RefluxNephropathy=No, pvDisease=No → ESRD-PolycsticKD=No, supp=0.77, conf=0.92
ESRD-RefluxNephropathy=No, pvDisease=No → ESRD-Renovascular=No, supp=0.77, conf=0.91
ESRD-RefluxNephropathy=No, pvDisease=No → lungDisease=No, supp=0.77, conf=0.91
ESRD-RefluxNephropathy=No, race=Caucasoid → cerebrovascular=No, supp=0.72, conf=0.89
ESRD-RefluxNephropathy=No, race=Caucasoid → ESRD-Analgesicnephropathy=No, supp=0.72, conf=0.93
ESRD-RefluxNephropathy=No, race=Caucasoid → ESRD-Diabetes=No, supp=0.72, conf=0.83
ESRD-RefluxNephropathy=No, race=Caucasoid → ESRD-PolycsticKD=No, supp=0.72, conf=0.91
ESRD-RefluxNephropathy=No, race=Caucasoid → ESRD-Renovascular=No, supp=0.72, conf=0.86
ESRD-RefluxNephropathy=No, race=Caucasoid → lungDisease=No, supp=0.72, conf=0.88
ESRD-RefluxNephropathy=No, race=Caucasoid → pvDisease=No, supp=0.72, conf=0.80
ESRD-Renovascular=No, hypertension=Yes → cancer=N, supp=0.70, conf=0.80
ESRD-Renovascular=No, hypertension=Yes → cerebrovascular=No, supp=0.70, conf=0.91
ESRD-Renovascular=No, hypertension=Yes → ESRD-Analgesicnephropathy=No, supp=0.70, conf=0.94
ESRD-Renovascular=No, hypertension=Yes → ESRD-PolycsticKD=No, supp=0.70, conf=0.92
ESRD-Renovascular=No, hypertension=Yes → ESRD-RefluxNephropathy=No, supp=0.70, conf=0.95
ESRD-Renovascular=No, hypertension=Yes → lungDisease=No, supp=0.70, conf=0.90
ESRD-Renovascular=No, hypertension=Yes → pvDisease=No, supp=0.70, conf=0.83
ESRD-Renovascular=No, lungDisease=No → cerebrovascular=No, supp=0.79, conf=0.92
ESRD-Renovascular=No, lungDisease=No → ESRD-Analgesicnephropathy=No, supp=0.79, conf=0.92
ESRD-Renovascular=No, lungDisease=No → ESRD-PolycsticKD=No, supp=0.79, conf=0.92
ESRD-Renovascular=No, lungDisease=No → ESRD-RefluxNephropathy=No, supp=0.79, conf=0.94
ESRD-Renovascular=No, pvDisease=No → cerebrovascular=No, supp=0.79, conf=0.85
ESRD-Renovascular=No, pvDisease=No → ESRD-Analgesicnephropathy=No, supp=0.74, conf=0.95
ESRD-Renovascular=No, pvDisease=No → ESRD-PolycsticKD=No, supp=0.74, conf=0.94
ESRD-Renovascular=No, pvDisease=No → ESRD-RefluxNephropathy=No, supp=0.74, conf=0.92
G.2. EXPERIMENT II - ASSOCIATION RULES

ESRD-Renovascular=No, pvDisease=No → ESRD-RefluxNephropathy=No, supp=0.74, conf=0.94
ESRD-Renovascular=No, pvDisease=No → lungDisease=No, supp=0.74, conf=0.92
ESRD-Renovascular=No, race=Caucasoid → cerebrovascular=No, supp=0.66, conf=0.91
ESRD-Renovascular=No, race=Caucasoid → ESRD-Analgesicnephropathy=No, supp=0.66, conf=0.92
ESRD-Renovascular=No, race=Caucasoid → ESRD-Diabetes=No, supp=0.66, conf=0.81
ESRD-Renovascular=No, race=Caucasoid → ESRD-PolycsticKD=No, supp=0.66, conf=0.91
ESRD-Renovascular=No, race=Caucasoid → ESRD-RefluxNephropathy=No, supp=0.66, conf=0.94
ESRD-Renovascular=No, race=Caucasoid → lungDisease=No, supp=0.66, conf=0.90
hypertension=Yes, lungDisease=No → cancer=N, supp=0.71, conf=0.80
hypertension=Yes, lungDisease=No → cerebrovascular=No, supp=0.71, conf=0.91
hypertension=Yes, lungDisease=No → ESRD-Analgesicnephropathy=No, supp=0.71, conf=0.95
hypertension=Yes, lungDisease=No → ESRD-PolycsticKD=No, supp=0.71, conf=0.93
hypertension=Yes, lungDisease=No → ESRD-RefluxNephropathy=No, supp=0.71, conf=0.95
hypertension=Yes, lungDisease=No → ESRD-Renovascular=No, supp=0.71, conf=0.88
hypertension=Yes, pvDisease=No → cerebrovascular=No, supp=0.64, conf=0.94
hypertension=Yes, pvDisease=No → ESRD-Analgesicnephropathy=No, supp=0.64, conf=0.95
hypertension=Yes, pvDisease=No → ESRD-PolycsticKD=No, supp=0.64, conf=0.92
hypertension=Yes, pvDisease=No → ESRD-RefluxNephropathy=No, supp=0.64, conf=0.95
hypertension=Yes, pvDisease=No → ESRD-Renovascular=No, supp=0.64, conf=0.90
hypertension=Yes, pvDisease=No → lungDisease=No, supp=0.64, conf=0.92
hypertension=Yes, race=Caucasoid → cerebrovascular=No, supp=0.60, conf=0.89
hypertension=Yes, race=Caucasoid → ESRD-Analgesicnephropathy=No, supp=0.60, conf=0.93
hypertension=Yes, race=Caucasoid → ESRD-Diabetes=No, supp=0.60, conf=0.82
hypertension=Yes, race=Caucasoid → ESRD-PolycsticKD=No, supp=0.60, conf=0.92
hypertension=Yes, race=Caucasoid → ESRD-RefluxNephropathy=No, supp=0.60,
APPENDIX G. PRELIMINARY ANALYSIS OF ANZDATA

conf=0.95
hypertension=Yes, race=Caucasoid → ESRD-Renovascular=No, supp=0.60, conf=0.85
hypertension=Yes, race=Caucasoid → lungDisease=No, supp=0.60, conf=0.89
hypertension=Yes, race=Caucasoid → pvDisease=No, supp=0.60, conf=0.80
lungDisease=No, pvDisease=No → cerebrovascular=No, supp=0.74, conf=0.94
lungDisease=No, pvDisease=No → ESRD-Analgesicnephropathy=No, supp=0.74, conf=0.95
lungDisease=No, pvDisease=No → ESRD-Diabetes=No, supp=0.74, conf=0.81
lungDisease=No, pvDisease=No → ESRD-PolycysticKD=No, supp=0.74, conf=0.92
lungDisease=No, pvDisease=No → ESRD-RefluxNephropathy=No, supp=0.74, conf=0.94
lungDisease=No, race=Caucasoid → cerebrovascular=No, supp=0.67, conf=0.91
lungDisease=No, race=Caucasoid → ESRD-Analgesicnephropathy=No, supp=0.67, conf=0.94
lungDisease=No, race=Caucasoid → ESRD-Diabetes=No, supp=0.67, conf=0.83
lungDisease=No, race=Caucasoid → ESRD-PolycysticKD=No, supp=0.67, conf=0.91
lungDisease=No, race=Caucasoid → ESRD-RefluxNephropathy=No, supp=0.67, conf=0.94
lungDisease=No, race=Caucasoid → ESRD-Renovascular=No, supp=0.67, conf=0.88
lungDisease=No, race=Caucasoid → pvDisease=No, supp=0.67, conf=0.83
pvDisease=No, race=Caucasoid → cerebrovascular=No, supp=0.62, conf=0.94
pvDisease=No, race=Caucasoid → diabetes=No, supp=0.62, conf=0.83
pvDisease=No, race=Caucasoid → ESRD-Analgesicnephropathy=No, supp=0.62, conf=0.93
pvDisease=No, race=Caucasoid → ESRD-Diabetes=No, supp=0.62, conf=0.88
pvDisease=No, race=Caucasoid → ESRD-PolycysticKD=No, supp=0.62, conf=0.91
pvDisease=No, race=Caucasoid → ESRD-RefluxNephropathy=No, supp=0.62, conf=0.93
pvDisease=No, race=Caucasoid → ESRD-Renovascular=No, supp=0.62, conf=0.90
pvDisease=No, race=Caucasoid → lungDisease=No, supp=0.62, conf=0.91
Author/s:
KUO, YEN-TING

Title:
Mining surprising patterns

Date:
2009

Citation:

Publication Status:
Unpublished

Persistent Link:
http://hdl.handle.net/11343/35312

File Description:
Mining surprising patterns

Terms and Conditions:
Terms and Conditions: Copyright in works deposited in Minerva Access is retained by the copyright owner. The work may not be altered without permission from the copyright owner. Readers may only download, print and save electronic copies of whole works for their own personal non-commercial use. Any use that exceeds these limits requires permission from the copyright owner. Attribution is essential when quoting or paraphrasing from these works.